NASA Technical Paper 1565

Algorithm for Fixed-Range Optimal Trajectories

Homer Q. Lee and Heinz Erzberger

JULY 1980

NASA

NASA Technical Paper 1565

Algorithm for Fixed-Range Optimal Trajectories

Homer Q. Lee and Heinz Erzberger
Ames Research Center
Moffett Field, California

National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1980

NOMENCLATURE

A	speed of sound
C_D	drag coefficient
C_L	lift coefficient
C_L basic, C_{Lo}, $C_{L\alpha}$, C_L gear	components of lift coefficient
D	drag force
d	ground track distance into the flight, running variable
d_c	total cruise distance
d_t	range of synthesized trajectory
d_{up}, d_{dn}	total climb and descent distance
E	aircraft energy, ft
$\dot{E}$	aircraft energy rate, ft/sec
E_c	aircraft energy in cruise
E_c^*	cruise energy corresponding to the optimal cruise cost λ^*
E_{ci}	aircraft energy at beginning of cruise
E_f	aircraft energy at end of flight
E_i	aircraft energy at beginning of flight
E_j	jth energy level
$\dot{E}_j$	rate of energy change from E_j to E_{j+1}
EPR	engine pressure ratio
EPR_{idle}	throttle setting in EPR for idle thrust
EPR_{min}, EPR_{max}	throttle setting in EPR for minimum and maximum thrusts, respectively. Note that the thrust delivered by throttle setting from EPR_{min} to EPR_{max} is continuous; on the other hand, there is a discontinuity in thrust delivered by throttle setting at EPR_{idle} and EPR_{min} (see fig. 4)
F_c	cruise fuel

$\tilde{F}_c$	estimated cruise fuel
F_n	corrected thrust
F_{up}	climb fuel
f	fuel consumption, lb
f_c	fuel cost, $/lb
$\dot{f}$	fuel flow rate, lb/hr
$\dot{f}_j$	fuel flow rate associated with energy level E_j
g	acceleration of gravity 32.2 ft/sec^2
H	Hamiltonian
h	aircraft altitude, a running variable, ft
h_c	cruise altitude
$h_{ceiling}$	ceiling altitude or the altitude at which the maximum thrust curve is tangent to the drag curve
h_f	aircraft altitude at end of flight
h_i	aircraft altitude at the beginning of flight
h_j	altitude resulting from minimizing the Hamiltonian for the jth energy level
h^*	optimal cruise altitude
$h^{\#}$	quantized altitude closest to optimal cruise altitude h^*; level of quantization is 1000 ft
h_{min}, h_{max}	lower and upper boundaries for altitude in cruise optimization, respectively
$h_{min,j}$, $h_{max,j}$	minimum altitude which prevents the aircraft from losing altitude in climb and the maximum altitude which prevents the aircraft from gaining altitude in descent for the jth evergy level. These two limits are translated into speed limits $V_{max,j}$ and $V_{min,j}$ respectively via the energy equation
I_{up}, I_{dn}	components of Hamiltonian
$I(V)$	cost function minimized over airspeed V

$I(\pi)$	cost function minimized over throttle setting π
I^*	optimal cost, a global minimum
I_c^*	optimal cost, minimized over the speed V and the range of EPR setting from EPR_{min} to EPR_{max}
I_{id}^*	optimal cost, minimized over speed at idle thrust
J	cost function
KC	temperature correction for fuel flow rate
L	lift force
M	Mach number
P	integrand of cost function = $f_c\dot{f} + t_c$
P	static pressure at given altitude
P_o	static pressure at sea level
R	specified range of trajectory
R^*	range uniquely specified by λ_{max}
R_{min}	range uniquely specified by λ_{min}
R_t	range of synthesized trajectory
S	wing surface area
T	ambient temperature, K
T	thrust, lb
TSFC	thrust specific fuel consumption
T_a	total air temperature, K; it is a function of altitude and ambient temperature
T_{max}	maximum thrust
T_{min}	minimum thrust
T_o	temperature at sea level, K
t	time into the flight, a running variable
t_c	time cost, \$/hr

t_c	time at the end of climb
t_d	time at the beginning of descent
t_f	time at the end of flight
V_f	airspeed at end of flight
$V_{g,j}$	groundspeed associated with energy level E_j
V_i	airspeed at beginning of flight
$V_{max,h,j}$	maximum airspeed resulting from altitude restrictions from the jth energy level (see $h_{max,j}$)
$V_{max,j}$	maximum airspeed for the jth energy level
$V_{max,opr}$	maximum airspeed from aircraft operational consideration
$V_{max,TD}$	maximum airspeed from thrust drag considerations. For cruise optimization, it is the intersection of the $T_{max} \cos \alpha$ versus drag curve. For climb and descent optimization, it is the intersection of the appropriate T versus the drag curve
$V_{min,c}$	lower speed limit as a result of altitude limit which keeps the aircraft from overshooting the cruise altitude h_c in climb
$V_{min,h,j}$	minimum airspeed resulting from altitude restrictions from the jth energy level (see $h_{min,j}$)
$V_{min,j}$	minimum airspeed for the jth energy level
$V_{min,opr}$	minimum airspeed from aircraft operational consideration
$V_{min,TD}$	minimum airspeed from thrust drag considerations. For cruise optimization, it is the intersection of the $T_{max} \cos \alpha$ versus drag curve. For climb and descent optimization, it is the intersection of the appropriate T versus the drag curve
v	true airspeed, running variable
v_g	groundspeed
v_{dn}	descent airspeed
v_{up}	climb airspeed
v_w	wind speed; function of altitude, running variable
v_{wdn}	wind speed in descent segment, function of altitude
v_{wup}	wind speed in climb segment, function of altitude

W	aircraft weight, lb
$\bar{W}$	average cruise weight
W_{cf}	aircraft weight at end of cruise
W_{ci}	aircraft weight at beginning of cruise
W_f	aircraft weight at end of flight
W_i	aircraft weight at beginning of flight
WFC	uncorrected fuel flow rate
W_x	x component of wind
W_y	y component of wind
$\partial W_x/\partial h$, $\partial W_y/\partial h$	x and y components of wind shear
x	ground track distance, running variable
x_{dn}	ground track distance for descent, running variable
x_{up}	ground track distance for climb, running variable
α	angle of attack
γ	ratio of specific heat = 1.4
γ	flightpath angle
$\dot{\gamma}$	rate of change of flightpath angle
γ_j	flightpath angle associated with energy level E_j
Δd_j	distance traversed, flying from energy level E_j to energy level E_{j+1}
Δf_j	fuel consumption, flying from energy level E_j to energy level E_{j+1}
Δh_j	difference between altitudes associated with energy level E_j and E_{j+1} respectively
Δt_j	time to flight from energy level E_j to energy level E_{j+1}
δ	air pressure ratio P/P_o
δ_a	pressure ratio corrected for Mach number effect

θ	temperature ratio T/T_o
λ	adjoint variable, cruise cost in \$/nm (a local minimum) E_{j+1}
$\bar{\lambda}$	average cruise cost
λ^*	optimal cruise cost, a global minimum
λ^*_{cf}	optimal cruise cost at end of cruise
λ^*_{ci}	optimal cruise cost at beginning of cruise
λ_f	cruise cost at end of cruise
λ_i	cruise cost at beginning of cruise
λ_{max}	cruise cost corresponding to the lowest cruise energy. It is the maximum cruise cost for all practical purposes
λ_{min}	cruise cost corresponding to the highest cruise energy. It is the minimum cruise cost for all practical purposes
π	throttle setting
π_{dn}	throttle setting in descent
π_{up}	throttle setting in climb
ρ	air density
$\rho_{s\ell}$	air density at sea level and standard pressure (29.92 in Hg) and temperature (15° C)
ϕ	bank angle
ψ	aircraft heading
$\%_\lambda$	percentage increase of λ above λ^*

ALGORITHM FOR FIXED-RANGE OPTIMAL TRAJECTORIES

Homer Q. Lee and Heinz Erzberger

Ames Research Center

SUMMARY

An algorithm for synthesizing optimal aircraft trajectories for specified range has been developed and implemented in a computer program written in FORTRAN IV. This report describes the algorithm, its computer implementation, and a set of example optimum trajectories for the Boeing 727-100 aircraft.

The algorithm optimizes trajectories with respect to cost function that is the weighted sum of fuel cost and time cost. Minimum fuel and minimum time trajectories are obtained for special cases of this cost function. Proper selection of the weighting factors gives minimum direct operating cost. The user has the choice of two modes of climb and descent optimization: optimizing over airspeed, or optimizing over both airspeed and thrust.

The algorithm is based on the assumption that there are at most three segments to the trajectories — climb, cruise, and descent. This results in one of the following three types of trajectories: (1) climb and descent; (2) climb, cruise at an altitude below the optimum cruise altitude, and (3) climb, cruise at optimum cruise altitude, and descent. Type (1) arises in short-range flights of 300 n. mi. or less, when optimizing over airspeed and thrust. Type (2) also arises in short-range flights, but only when thrust is constrained to a maximum value in climb and to a minimum value in descent. Type (3) arises in long-range flights. The climb and descent profiles are generated by integrating a simplified set of kinematic and dynamic equations, wherein the total energy of the aircraft is the independent or time-like variable. At each energy level, the optimum airspeeds and thrust settings are obtained as the values that minimize the variational Hamiltonian.

The minimization is carried out by the Fibonacci search technique. The Hamiltonian also depends on an adjoint variable which is shown to be identical to the optimum cruise cost at a specified altitude. The adjoint variable not only specifies cruise condition but also, indirectly, the climb and descent profiles and therefore the range. In general, the functional relationship between the adjoint variable and range cannot be determined explicitly; therefore, iteration is used to achieve a specified range.

An important part of the algorithm involves correcting the trajectories for weight loss due to fuel burn. Both the adjoint variable and the energy rate are corrected during climb and descent to account for the change in weight.

The computer implemented version of the algorithm was used to compute and evaluate the characteristics of several types of optimum trajectories. In these computations the aerodynamic and propulsion models used are

representative of the Boeing 727-100 equipped with Pratt and Whitney JT8D-7A engines. In comparison with minimum-cost trajectories, minimum-fuel trajectories reach higher altitudes, using a steeper flightpath angle in climb and a shallower one in descent. For the chosen values of fuel cost and time cost, the minimum-fuel trajectories for specified range reduce fuel consumption about 1 lb/n. mi. and increase the flight time 8 min/hr flown, with respect to the minimum operating cost trajectories. Differences in fuel consumption and operating cost between climb and descent profiles optimized with respect to airspeed only and optimized with respect to both airspeed and thrust, were found to be slight, though these differences are strongly model dependent.

The computer program and its use are described in the appendix.

INTRODUCTION

The continuing rise in airline operating costs due to higher fuel prices and other inflationary factors has stimulated considerable interest in reducing costs by trajectory optimization. A simplified analytical approach for calculating cost-optimum trajectories for airline missions was described in references 1 and 2. The optimum trajectories calculated by this approach, unlike those obtained by classical quasi-static performance analysis, minimize an integral performance measure, such as total mission fuel. Although the calculations prescribed by the approach are based on reduced-state dynamic models, they are still too complex for hand calculations and must be done on a digital computer. This report first summarizes the analytical approach and then develops a computational algorithm, implemented in a FORTRAN program, for generating the optimum trajectories.

There are several uses for a computer implementation of such an algorithm. First, it can serve as a research tool for investigating various characteristics of optimum trajectories. For example, it can determine attainable reduction in fuel consumption and operating costs by flying an optimum rather than a standard reference trajectory. Second, it can be used to generate a model for an airborne computer implementation. Third, it can be incorporated into an airline flight planning system. In this latter application, it can provide an optimum flight plan for a specific airline mission for conditions prevailing at time of takeoff. Although computerized flight-planning systems are popular with many airlines, they do not now produce optimum trajectories. They also do not produce flight plans for short-haul routes, where the potential for cost savings has been shown to be significant (ref. 3).

The approach on which the algorithm is based assumes that trajectories consist of climb, cruise, and descent segments. However, the optimization of each segment is not done independently as in classical procedures but specifically accounts for interaction between segments. This is accomplished by application of optimal control theory, as described in references 1 and 2.

The report is organized as follows: The first section defines the performance function, the dynamical equations of flight and the various optimization options available. In the second section, the function-minimization

method is described and the manner in which the boundaries of the region over which the function is to be minimized are established. The third section gives the procedure for estimating the weights at cruise and landing in order to optimize the climb and descent. The fourth section describes how the program generates a trajectory with specified range. The last section discusses the characteristics of several example fuel- and cost-optimum trajectories computed by the program.

An appendix contains the user's manual for the program. A brief description is given of all subroutines and the input data necessary for specifying a desired trajectory. The aircraft model data incorporated in the program is a reasonable approximation to the lift, drag, and propulsion characteristics of the Boeing 727-100 aircraft. The algorithm's performance has been extensively exercised with this model by computing trajectories for large variations in range, weight, winds, and temperatures. The program is structured to minimize the difficulty of replacing the 727-100 aircraft model with models of other aircraft.

ANALYTICAL FORMULATION

In this section, the cost function and the essential equations for synthesizing optimal trajectories are given. Since they are derived in reference 1, they are simply stated without proof.

The cost function to be minimized consists of the sum of the fuel cost and the time cost,

$$J = f_c f + t_c t \tag{1}$$

where f_c and t_c are the unit cost of fuel and time, respectively. In integral form,

$$J = \int_0^{t_f} (f_c \dot{f} + t_c)dt \equiv \int_0^{t_f} P\, dt \tag{2}$$

where t_f is the time of the flight.

The optimal trajectory, which minimizes the cost function, is assumed to consist of three segments, a climb, a steady cruise, and a descent segment. Rewriting equation (2) in three segments results in

$$J = \underbrace{\int_0^{t_c} P\, dt}_{\text{climb cost}} + \underbrace{(R - d_{up} - d_{dn})\lambda}_{\text{cruise cost}} + \underbrace{\int_{t_d}^{t_f} P\, dt}_{\text{descent cost}} \tag{3}$$

where λ designates the cost per mile at cruise, R the specified range (ground distance of the flight) and d_{up} and d_{dn} are the distances covered during climb and descent, respectively.

The aircraft energy is defined as

$$E = h + (1/2g)V^2 \tag{4}$$

and the energy rate $\dot{E}$ is obtained from the relation,

$$\dot{E} = (T - D)V/W \tag{5}$$

with the constraint that lift is equal to weight.

Next, changing from time to energy as the independent variable in equation (3) by the transformation $dt = dE/\dot{E}$, yields

$$J = \int_{E_i}^{E_c} (P/\dot{E}]_{\dot{E}>0})dE + (R - d_{up} - d_{dn})\lambda + \int_{E_f}^{E_c} (P/|\dot{E}|]_{\dot{E}<0})dE \tag{6}$$

where

E_c cruise energy

E_i initial climb energy

E_f final descent energy

This transformation assumes that the trajectory consists of at most three segments — climb, cruise, and descent — characterized by montonically increasing, constant, and montonically decreasing energy, respectively. This places strict inequality constraints on $\dot{E}$. Also, the integration limits in the descent cost term have been reversed for convenience.

In order to calculate the distance flown along a trajectory, we need the differential equation for distance,

$$dx/dt = V \cos \gamma + V_w \sim V + V_w \tag{7}$$

where the approximate relation can be (and was in the computer program) used for small γ, since the maximum flightpath angle γ for most transport aircraft is about 5°. This differential equation is also transformed by changing to energy as the independent variable and then separating it into climb and descent components,

$$d\,x_{up}/dE = (V_{up} + V_{wup})/\dot{E}]_{\dot{E}>0} \;, \quad d\,x_{dn}/dE = (V_{dn} + V_{wdn})/|\dot{E}|]_{\dot{E}<0} \tag{8}$$

We now have a problem of optimal control in which the state variable is $x_{up} + x_{dn}$, the control variables are V_{up}, V_{dn}, π_{up}, π_{dn}, and the independent or time-like variable is E.

One of the necessary conditions for the minimization of equation (6) subject to the state of equation (8) is

$$\operatorname*{Min}\, H\,(E,\ \lambda) = \operatorname*{Min}_{\substack{V_{up}, V_{dn}\\ \pi_{up}, \pi_{dn}}} \left\{ \frac{P}{\dot{E}}_{\dot{E}>0} + \frac{P}{|\dot{E}|}_{\dot{E}<0} + \lambda \left[\frac{V_{up} + V_{wup}}{\dot{E}|_{\dot{E}>0}} + \frac{V_{dn} + V_{wdn}}{|\dot{E}|_{\dot{E}<0}} \right] \right\} \tag{9}$$

where H is the Hamiltonian. The quantity λ is an adjoint variable which is identical to the cost of cruising at an energy E_c. By distributing the minimization operator in equation (9), H can be decomposed into climb and descent components,

$$\operatorname{Min}\, H = I_{up} + I_{dn} \tag{10}$$

where

$$I_{up} = \operatorname*{Min}_{\substack{V_{up}\\ \pi_{up}}} \left[\frac{P - \lambda(V_{up} + V_{wup})}{\dot{E}]_{\dot{E}>0}} \right], \quad I_{dn} = \operatorname*{Min}_{\substack{V_{dn}\\ \pi_{dn}}} \left[\frac{P - \lambda(V_{dn} + V_{wdn})}{|\dot{E}|]_{\dot{E}<0}} \right] \tag{11}$$

The optimum climb and descent profiles are defined by the functional relation between the independent variable energy and the dependent variables, airspeed, and power setting. These relations are obtained by performing the minimization in equation (11) for every climb and descent energy.

Knowing the initial airspeed and altitude and the final airspeed and altitude, the initial climb and descent energies are specified through equation (4). The cruise-cost λ, for a specified cruise energy E_c, is generated from the following relation,

$$\lambda = \operatorname*{Min}_{V} \frac{f_c \dot{f} + t_c}{V + V_w} \tag{12}$$

with trim constraints

$$T(\pi,\ V,\ h)\cos\alpha - D = 0 \tag{14}$$

$$T(\pi,\ V,\ h)\sin\alpha + L - W = 0 \tag{15}$$

where the thrust T is a function of power setting π, the airspeed V, and the altitude h; L is the lift. Equations (14) and (15) are equations of motion along the wind and lift axes for constant-speed level flight.

A climb profile is generated by computing the V_{up} and π_{up} which minimizes I_{up} for a given aircraft-energy E, from E_i to E_f. Specifically, since E is monotonically increasing, the profile starts from E_i, increments by ΔE and finishes at E_c. Similarly, the descent profile is generated by computing the V_{dn} and π_{dn}, which minimizes I_{dn} for a given energy E, starting from E_f and ending at E_c. In the climb segment (and analogously, in the descent segment), the time duration Δt_j for any energy increment ΔE_j, the distance into the climb d, the flightpath angle γ_j, and the cumulative fuel consumption F are computed from

$$\Delta t_j = \Delta E/\dot{E}_j \tag{16}$$

$$d = \Sigma \Delta d_j, \quad \Delta d_j = V_{gj} \cos \gamma_j \Delta t_j \tag{17}$$

$$\sin \gamma_j = (\Delta h_j/\Delta t_j)/V_{gj} \tag{18}$$

$$F = \Sigma \Delta F_j, \quad \Delta F_j = \dot{f}_j \Delta t_j \tag{19}$$

where the subscript j refers to energy level $E_j (E_i \le E_j \le E_c$ for climb, and $E_f \le E_j \le E_c$ for descent); Δh_j is the difference in altitude between energy level E_{j-1} and E_j; and V_{gj} is the groundspeed corresponding to the airspeed V at energy level E_j.

A trajectory with fixed range R must satisfy the constraint

$$d_{up} + d_{dn} \le R \tag{20}$$

Since the range is the sum of the climb distance, the descent-distance, and the cruise distance (if there is a cruise segment), and since all three distances depend on E_c, the cruise energy uniquely specifies the range. The cruise energy and the cruise cost (per mile) λ are interdependent, and knowing the value of one completely specifies the value of the other. It is computationally expedient to use λ instead of E_c to specify the range. In other words, the trajectory with a specified range R is synthesized by varying the value of λ until the sum of the three ground-track segments falls within a small error of R. A typical functional dependence between λ and E_c is illustrated in figure 1.

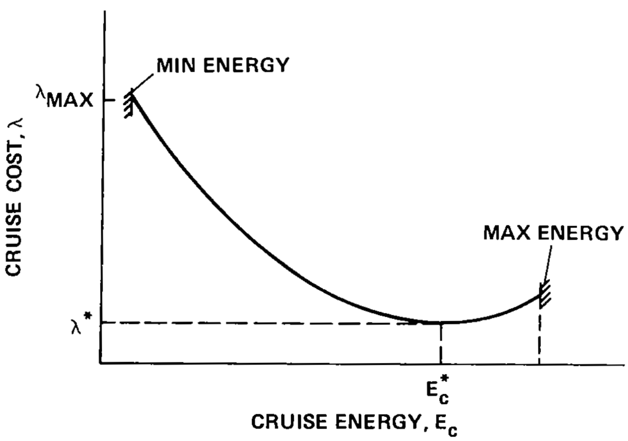


Figure 1.- Cruise cost function.

The range of allowable value of starts with λ^* and ends with λ_{max} for short-distance trajectories. The optimal cruise-cost λ^* is the result of minimizing equation (12) for all cruise energies, i.e.,

$$\lambda^* = \underset{E_c}{\mathrm{Min}}\ \underset{V}{\mathrm{Min}}\ \frac{f_c\dot{f} + t_c}{V + V_w} \tag{21a}$$

This also gives the optimal cruise altitude h^* via equation (4). The optimal cruise cost λ^* occurs near the maximum or ceiling altitude for a given aircraft weight. The cruise energy corresponding to λ^* will be designated by E_c^*. The quantity λ_{max} is chosen to correspond to the lowest cruise energy of interest for commercial flight.

Computationally, it is convenient to carry out the second minimization over altitude h instead of over E_c. Since V is fixed after the first minimization, E_c is obtained by use of equation (4) or known h. Equation (21a) can be rewritten, therefore, as

$$\lambda^* = \underset{h}{\mathrm{Min}}\ \underset{V}{\mathrm{Min}}\ \frac{f_c\dot{f} + t_c}{V + V_w} \tag{21b}$$

$$h^*, V^* = \arg \underset{h,V}{\mathrm{Min}}\ \lambda^* \tag{21c}$$

There are two modes of climb and descent optimization incorporated in the program. In the constrained-thrust mode, climb and descent are optimized with respect to airspeed, with power held at some maximum value in climb and at idle in descent. In the free-thrust mode, both airspeed and power are optimized in climb and descent. These two modes will generate different types of trajectories as the range is varied. In the former, the trajectories generally will contain cruise segment even for short-range flight, whereas in the latter no cruise segment generally occurs until the range is of sufficient length to permit the aircraft to climb to and descend from the optimum cruise energy. The length of the cruise segment for the short-range, constrained-thrust mode is obtained by a trial or iterated value of λ. As shown in reference 1, this length is obtained from the relation,

$$d_c = - \left. \frac{I_{up} + I_{dn}}{d\lambda/dE} \right|_{E=E_c} \tag{22}$$

Depending on the specified range, these modes result in one of the following three types of trajectories: (1) climb-descent; (2) climb, cruise at an energy below E_c^*, descent (referred to as climb-cruise-descent); and (3) climb to optimal cruise energy E_c^*, cruise at E_c^*, and then descent from E_c^* (referred to as climb-cruise*-descent). The range (and also the cruise altitude or cruise energy in the second case) increases with decreasing value of λ, starting from λ_{max} to λ. The ranges corresponding to λ_{max} and λ^* are denoted by R_{min} and R^*, respectively. If the specified range R is greater than R^*, then the trajectory consists of climb-cruise*-descent segments with a cruise distance equal to $R - (d_{up} + d_{dn})$. If R is between

R_{min} and R*, then the appropriate λ which yields the desired range must be chosen by iteration. The iterated value of λ is expressed as $\%_\lambda$, the percentage increase of λ above λ^*, i.e.,

$$\%_\lambda = (\lambda - \lambda^*)(100)/\lambda^* \tag{23}$$

In the constrained thrust mode, equation (22) is used to compute the cruise distance for any $\%_\lambda \geq 1\%$. In the free-thrust mode, the cruise distance is taken as zero for any $\%_\lambda \geq 1\%$. For $\%_\lambda \geq 1\%$, the program assumes cruise to occur at optimum cruise energy. Numerical stability dictated the choice of 1% (instead of 0%, i.e., at λ^*) as the end point for equation (22). The family of trajectories obtained for the two modes of optimization are illustrated in figure 2.

Figure 3 shows the percent increase of λ versus range for a

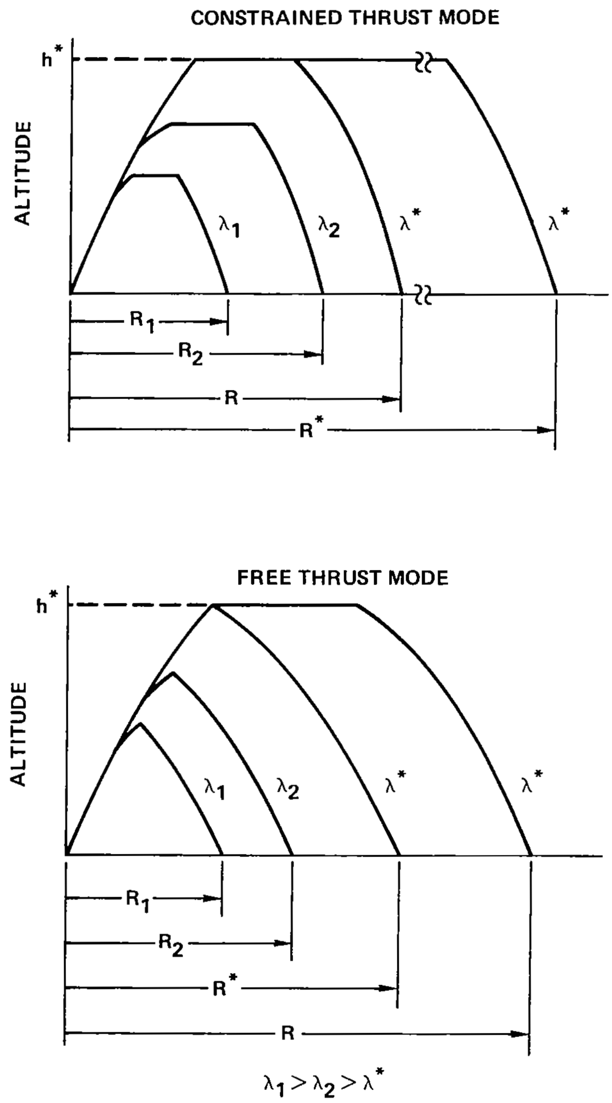


Figure 2.- Altitude-range profile for different values of λ.

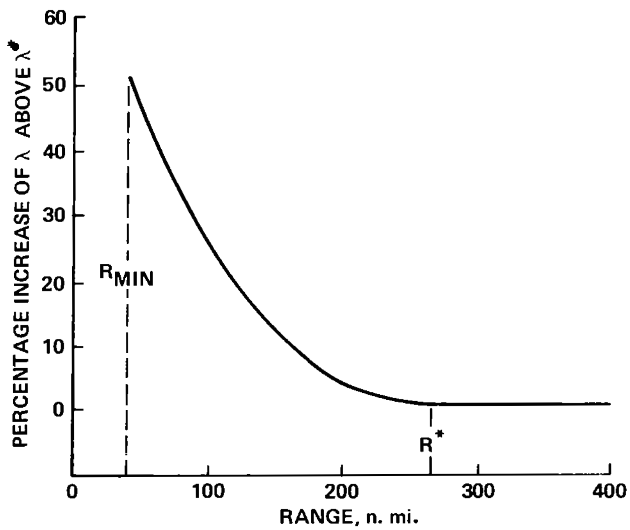


Figure 3.- Cruise cost in $\%_\lambda$ vs range.

particular case, an initial aircraft weight of 136,000 lb and minimum cost trajectories. The slope of the curve is very steep at short range (from R_{min} to 130 n. mi.), becomes shallow at 200 n. mi., and is constant at ranges above R^*. These properties of the curve are incorporated in the algorithm for iterating on λ.

Table 1 summarizes the types of trajectories resulting from all combinations of modes of optimization and ranges in relation to R^*.

TABLE 1.- RANGE, CRUISE COST, OPTIMIZATION MODE, AND TRAJECTORY TYPE

λ	Mode[a]	R	Trajectory type
$> \lambda^*$	CT	$< R^*$	Climb-cruise-descend
$> \lambda^*$	FT	$< R^*$	Climb-descend
$= \lambda^*$	CT	$= R^*$	Climb-cruise*-descend
$= \lambda^*$	FT	$= R^*$	Climb-descend
$= \lambda^*$	CT	$> R^*$	Climb-cruise*-descend
$= \lambda^*$	FT	$> R^*$	Climb-cruise*-descend

[a]CT = constrained thrust; FT = free thrust.

TECHNIQUE OF MINIMIZATION

The effectiveness of the trajectory optimization program depends strongly on the numerical procedure for finding the minimum of a function of one or two variables. Specifically, for cruise optimization, the cost, as expressed in equation (21b), is minimized with respect to both altitude h and true airspeed V. For climb and descent optimization, the Hamiltonian function, as expressed in equation (11), is minimized with respect to both true airspeed and thrust.

Fibonacci search (ref. 4) is used for minimization of these functions. The advantage of using Fibonacci search is that of all known search techniques it requires the least number of function evaluations to locate the minimum with prescribed accuracy. Also, the technique is suitable for handling tabulated data since no function differentiation is required and since it allows

discontinuities in the function. Furthermore, no a priori knowledge of the function is required except that it be unimodel.

A prerequisite for starting the search is that the lower and upper boundaries of the search region be well defined. The numerical accuracy of locating the minimum within this region depends on the range of the search region and highest Fibonacci number used; specifically, the accuracy is the quotient of the range and the highest Fibonacci number.

Fibonacci search is basically a one variable minimization procedure. It is adapted here for minimization with respect to two variables by applying the technique to one variable at a time while holding the second variable fixed. Convergence to the minimum is achieved by cycling between the two variables as required.

Application of this technique for cruise-performance optimization and for climb and descent profile optimization will be discussed in that order.

Cruise Performance Optimization

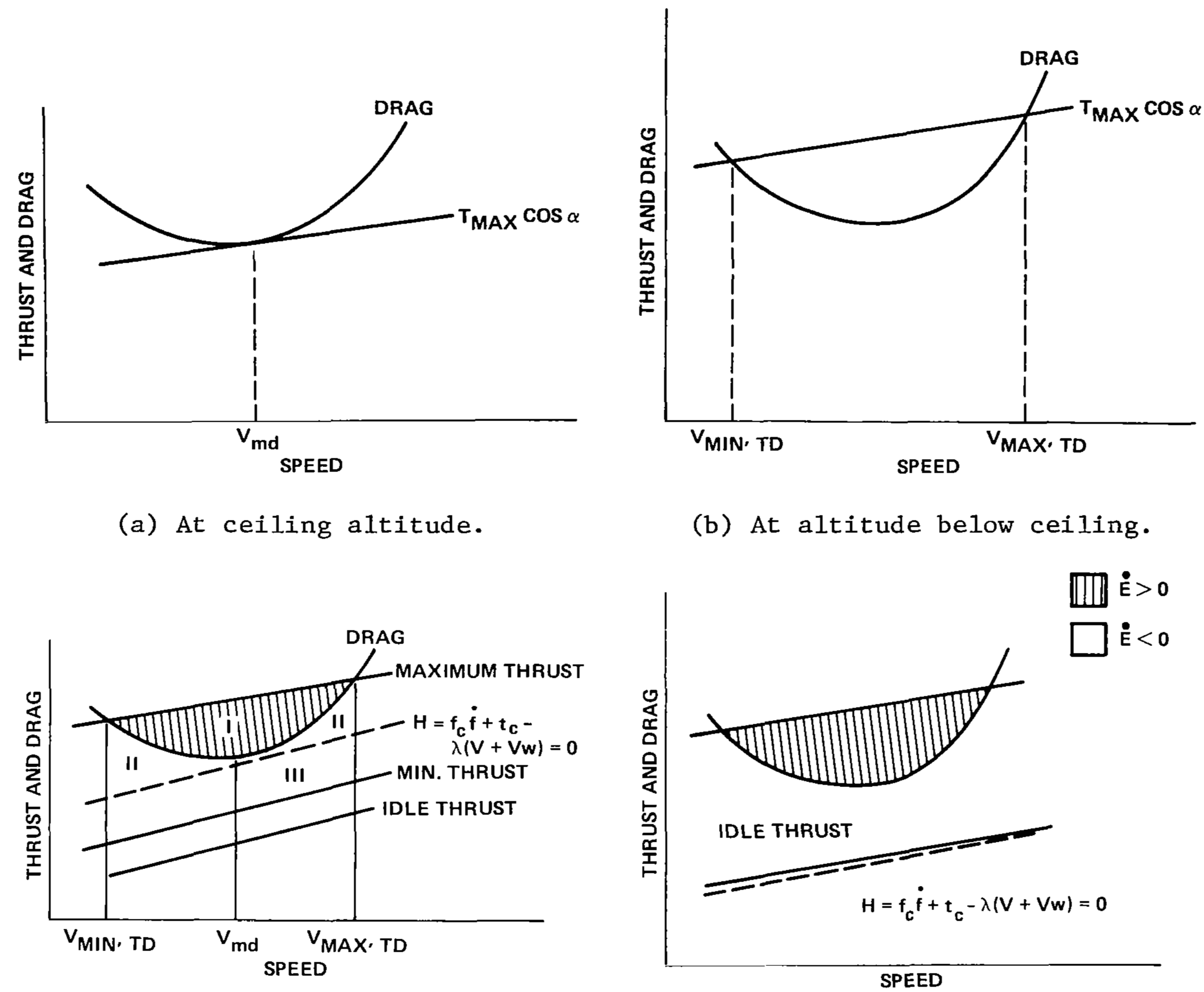
In the cruise performance optimization, the cost is minimized with respect to both h and V simultaneously. Since cruise can occur at altitudes below the optimum cruise altitude, the program is set up to generate the optimum cruise speed, cruise cost, and power setting at selected altitude intervals. The altitude range for cruise optimization starts from some preselected lowest altitude, usually at sea level, and ends at the ceiling altitude.

The ceiling altitude is the maximum altitude at which the aircraft can be kept in trim; that is, the highest altitude at which equations (14) and (15) are satisfied. This altitude is the upper bound of the altitude range for cruise performance optimization.

At the ceiling, $T_{max} \cos \alpha$ is equal to drag. At any lower altitude $T_{max} \cos \alpha$ is greater than drag. Figures 4(a) and 4(b) illustrate this relation. Figure 4(a) shows that a ceiling altitude, $T_{max} \cos \alpha$ is tangent to the drag curve and figure 4(b) shows that below the ceiling, the maximum thrust and the drag curves cross each other at two speeds, denoted by $V_{min,TD}$ and $V_{max,TD}$, respectively. They are the two speed limits from thrust-drag consideration. At any speed between the two,

$$T_{max} \cos \alpha > D \tag{24}$$

and there is always enough thrust to overcome drag so that equation (24) can be satisfied. At low altitude, there is a relatively wide range of speed between $V_{min,TD}$ and $V_{max,TD}$. As the aircraft flies to higher altitudes, this speed range becomes smaller. At the ceiling altitude, there is only one speed at which equation (14) is satisfied, namely, the minimum drag speed V_{md}. Other constraints that may limit the speed range are the operating speed limits



(a) At ceiling altitude.

(b) At altitude below ceiling.

(c) At high altitude.

(d) At low altitude.

Figure 4.- Thrust and drag versus speed.

$V_{min,opr}$ and $V_{max,opr}$, respectively. Typical value of these limits for current transport aircraft are $M = 0.1$ and $M = 0.89$. Therefore, the lower bound for speed V_{min} is the greater of $V_{min,TD}$ and $V_{min,opr}$ and the upper bound V_{max} is the lesser of $V_{max,TD}$ and $V_{max,opr}$. The cruise performance is computed at various altitudes by minimizing the cost function, equation (12), over the speed range (V_{min}, V_{max}). The altitudes where the cost is computed are incremented at regular intervals, starting from sea level and ending at the ceiling altitude. The cruise cost exhibits the parabolic-like shape as a function of altitude shown in figure 1, with a well defined minimum. The minimum of the cruise cost with respect to the discrete altitudes establishes the location of the optimum cruise point to within one incremental interval. The location of the optimum is now computed more accurately by

applying the Fibonacci technique to the narrow altitude region, spanned by two incremental intervals and centered on the discrete altitude minimum.

An an example, consider a case in which the altitude is incremented by 1,000 ft and denote the discrete altitude at which the cost is minimum by $h\#$. Then the optimum is in the altitude region defined by (h_{min}, h_{max}), where

$$h_{min} = h\# - 1000, \qquad h_{max} = h\# + 1000 \tag{25}$$

provided

$$h_{max} \le h_{ceiling}$$

By applying a series of eight Fibonacci numbers to search for the optimum over this region, it can be located to within ±36.36 ft, since 2000/55 equals 36.36 ft, where 55 is the eighth Fibonacci number.

The accuracy of the location of the minimum can be improved by increasing the number of Fibonacci numbers. A tolerance of less than ±10 ft is usually not needed in practical applications. For simplicity, the nearly optimal altitude resulting from the Fibonacci search will be identified with the optimum cruise altitude h^*.

At the end of cruise optimization, a multidimensional array is constructed, consisting of an array of altitudes (typically at 1000-ft intervals at low altitudes and at smaller intervals in the vicinity of the ceiling altitude) together with corresponding arrays of optimal cruise speeds, power settings, cruise costs, fuel-flow rates, and energy heights, all at a fixed aircraft weight. The calculation is repeated at each aircraft weight of interest, giving optimal cruise altitude, optimal cruise cost, speed, power setting, fuel-flow rate, and energy height each as a function of weight. The totality of these arrays and their corresponding optimal quantities, one for each fixed aircraft weight, ranging from the heaviest possible weight to the lightest, typically in increments of 5000 lb, will be referred to as the cruise table. The data in this table are necessary for synthesizing optimal trajectories.

Climb and Descent Optimization

In the climb and descent profile optimization, the search is applied to two control variables in sequence, namely, speed and thrust. In this two-dimensional search, the search region is recomputed whenever the Fibonacci search is switched from one control variable to the other. This is necessary because the range of the search region generally depends on the value of the control variable that is being held fixed. The two-dimensional search consists of: (1) defining the range of the first control variable; (2) minimizing the function over the first control variable in the prescribed range; (3) using the value of the first control variable obtained in (2) to define the range of the second control variable; (4) minimizing the function over the second control variable within the range prescribed in (3); and (5) repeating the process until the controls have converged to the minimum.

Depending on the form of the engine data supplied by the manufacturer, the thrust T can be specified by either fan RPM, (N_1), or by engine pressure ratio (EPR). The corresponding maximum thrust is specified by $N_{1,max}$ or EPR_{max} and the corresponding idle thrust is specified by $N_{1,idle}$ or EPR_{idle}. In the remainder of this report, EPR will be used for thrust setting, since the thrust data supplied by the manufacturer for the example engine were given as a function of EPR.

The operating speed limits $V_{min,opr}$ and $V_{max,opr}$, which are common to both climb and descent optimization, together with other appropriate speed limits, are used to define the range of allowable speeds when minimizing the Hamiltonian given in equation (11). A second common speed limit, $V_{max,ATC}$ results from an ATC restriction that requires the maximum allowable speed at or below 10,000 ft to be limited to 250 KIAS.

Climb profile optimization- Derivation of the range of π_{up} and V_{up} for climb profile optimization will now be discussed. Figure 4(c) shows typical maximum thrust, drag, and idle thrust curves versus speed for a high altitude; figure 4(d) shows typical curves for a low altitude. The maximum and idle thrust curves give the thrust delivered by the engines at maximum EPR and idle EPR settings, respectively. The region enclosed by the maximum-thrust curve and the drag curve — the shaded region or region 1 — satisfies the strict inequality $T > D$ and is a valid region for speed and EPR setting optimization in climb.

Consider first the case in which the Fibonacci search is over thrust setting, EPR, as some fixed airspeed. The upper bound of the search is simply the maximum EPR, EPR_{max}, at the fixed airspeed. The lower bound is the value for which thrust equals drag at the fixed airspeed, i.e., the locus of $\dot{E} = 0$. Since $\dot{E}$ is the denominator of the Hamiltonian, the function becomes infinite at this EPR setting. To avoid numerical difficulties in this region, an EPR setting somewhat higher than EPR at $T = D$ is used in the program. This is accomplished by limiting the minimum value of $\dot{E}$ to a value slightly greater than zero, for example, to 5 ft/sec.

Next, consider the case in which the Fibonacci search is over airspeed at some fixed EPR. Plotting thrust vs airspeed at constant EPR setting traces out a curve similar to the maximum-thrust curve. The speeds at which this curve intersects the drag curve define some lower and upper limits for the speed range. In the case of the maximum-thrust curve, $V_{min,TD}$ and $V_{max,TD}$ define the limits from thrust-drag consideration. These two speeds will be used to determine the lower and upper bounds for the speed range.

In climb, the aircraft is not allowed to dive; therefore, the altitude defined for the current energy level h_j is considered as the minimum altitude for the computation at the next energy level, $h_{min,j+1}$. This altitude limit can be translated into an upper speed limit, at the $(j + 1)$th step, by energy equation (4), $V_{max,h,j+1} = \sqrt{2g(E_{j+1} - h_j)}$. Figure 5 shows the energy contours in the altitude speed plane and illustrates this constraint.

The speed at the end of climb is determined from the altitude ceiling, which prevents the aircraft from going above the cruise altitude. The

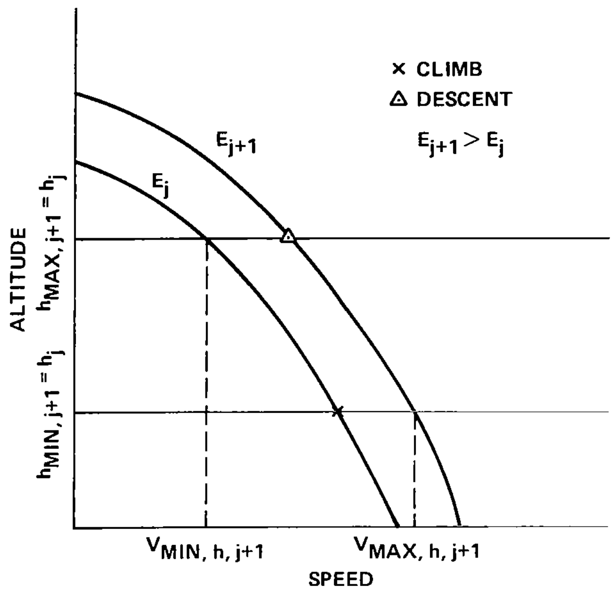


Figure 5.- Energy contours in the altitude-speed plane.

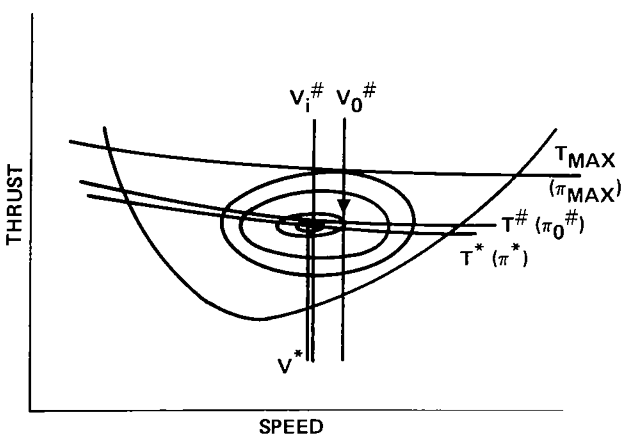


Figure 6.- Locus of Fibonacci search for minimal cost in climb optimization.

altitude ceiling is translated into a speed limit by the energy equation (4) $V_{min,c} = \sqrt{2g(E_j - h_c)}$.

The lower bound for speed at the jth step in the computations, $V_{min,j}$, is the greatest of $V_{min,TD}$, $V_{min,opr}$, and $V_{min,c}$. The upper bound $V_{max,j}$, is the least of $V_{max,TD}$, $V_{max,h}$, $V_{max,opr}$, and $V_{max,ATC}$ (the latter applies only if the aircraft is flying below 10,000 ft).

With the range of admissible EPR and speed well defined, the Hamiltonian function can now be minimized by application of Fibonacci search. For the constrained-thrust case, the search is over V only. The search interval is defined as from $V_{min,j}$ to $V_{max,j}$ along the maximum thrust line.

For the free-thrust case, the minimization is over V and T alternately. Figure 6 shows a typical sequence of free-thrust climb optimization. The first search is along the maximum-thrust line over the interval from V_{min} to V_{max} and minimizing over V. The resulting speed is denoted by $V_0^{\#}$. The next search is along $V_0^{\#}$, over the interval from EPR to EPR_{max}, minimizing the Hamiltonian over π. Denote the resulting EPR setting by $\pi_0^{\#}$. The next stage of search is along $\pi_0^{\#}$. over V again and this process is continued until the difference between two successive costs is within some small value, i.e.,

$$|I(V) - I(\pi)| \leq \varepsilon \tag{26}$$

where $I(V)$ is the cost resulting from minimization over V at constant T and $I(\pi)$ is the cost resulting from minimization over T at constant speed.

<u>Descent profile optimization</u>- The region of permissible π_{dn} and V_{dn} for descent profile optimization will be derived. The condition $\dot{E} < 0$ requires that thrust be less than drag; therefore, the complement of region I in figures 4(c) and 4(d) is the feasible region. It is possible to reduce the region to be searched by considering the sign of I_{dn} in equation (11). Since

I_{dn} is the quotient of $P - \lambda V$ and the absolute value of $\dot{E}$, the Hamiltonian takes the sign of the numerator. In region II — the region between locus of $H = 0$ (shown by dashed lines in figs. 4(c) and 4(d)) and the drag curve — I_{dn} is positive, since the numerator of H is positive. In region III — the region between the minimum-thrust curve (thrust with EPR setting at its minimum value) and the $H = 0$ line — I_{dn} is negative, since the numerator is negative. The value of I_{dn} in region III, a negative quantity, is always less than the value of I_{dn} in region II, a positive quantity. Therefore, region III is the region in which the minimum will be found.

At low altitude, it is possible for the $H = 0$ line to lie below the idle-thrust line. In this case the cost is positive and the minimum cost is located along the idle-thrust line. For a constant-speed search, the EPR setting ranges from idle EPR to EPR at $T = D$. To avoid having the Hamiltonian function become infinite at $T = D$, the upper bound of the EPR setting is chosen such that $\dot{E} < -5$ ft/sec.

For the constant EPR search, there is no speed limit from thrust-drag considerations. However, in descent, the aircraft is not allowed to gain altitude, and h_j, the altitude from the last energy level, is considered as the maximum altitude, $h_{max,j+1}$ for the next energy level. This altitude limit is translated into a lower speed limit via the energy equation $V_{min,h,j+1} = \sqrt{2g(E_{j+1} - h_j)}$. Figure 5 shows the relation between $h_{max,j+1} = h_j$ and $V_{min,h,j+1}$ at energy level E_j. Therefore, the lower bound for speed $V_{min,j}$ is the greater of $V_{min,h,j}$ and $V_{min,opr}$ and the upper bound for speed $V_{max,j}$ is $V_{max,opr}$ at altitudes above 10,000 ft and the lesser of $V_{max,opr}$ and $V_{max,ATC}$ (or 250 KIAS) below 10,000 ft.

With both EPR setting and speed ranges well defined, the Hamiltonian function for descent can now be minimized by application of Fibonacci search. In the unconstrained-thrust case, the first minimization is over π at constant V^*, where V^* is the optimal speed resulting from minimization at the last energy level. The iteration starts by selecting the final speed V_f for the first energy level E_f. The next step is to minimize the function over V and to repeat the process until the criterion for the minimum, as expressed in equation (26), is satisfied.

The data used in the computer program, as supplied by the manufacturer, typically have a discontinuity in thrust versus EPR, in that there is usually a gap between the lowest EPR setting for which thrust and fuel flow are tabulated and the idle-power setting. To account for this gap, the lower bound for EPR setting in region III is redefined as the lowest EPR setting curve, instead of the idle-power setting curve. The area of search is expanded to include both region III and the idle-power line. Denote by I_c^* and I_{id}^* the minimum values of cost obtained from minimizing the Hamiltonian over V and along the idle-thrust curve, respectively. Then the minimum value for descent optimization I^* is the lesser of the two values $I^* = \min(I_c^*, I_{id}^*)$. In the constrained-thrust case, the minimization is over V only along the specified thrust line, namely, the idle-thrust line.

To illustrate the numerical accuracy, consider the case in which the Fibonacci number is used for the search and the operating speed range is from

M = 0.1 to M = 0.9. The accuracy of the optimal speed obtained from the search is within (0.9 - 0.1)/144 = 0.0056 Mach number under the worst condition, where 144 is the 11th Fibonacci number. Such a large speed interval has not been yet encountered in practice; typically, the speed range is about half of this.

Typical values for EPR_{min} and EPR_{max} are 0.1 and 0.24, respectively. Therefore, the EPR setting is accurate to at least (2.4 - 1.1)/144 = 0.009 under the worst condition, since this EPR range is for both climb and descent and does not account for smaller maximum EPR value at higher altitude and higher temperature.

WEIGHT ESTIMATION

The Hamiltonian functions (eq. (11)) for climb and descent both contain λ, the cruise cost so that the value of λ must be known before one can generate the climb and descent profiles. The value of λ can be computed by rearranging terms in equation (23),

$$\lambda = \lambda^*[1 + (\%_\lambda/100)] \tag{27a}$$

where λ^* is the optimal cruise cost. The value of λ^* depends on the cruise weight. For trajectories containing a cruise segment, the initial cruise weight differs from the final cruise weight by the fuel burn during cruise. These two different weights result in different values of λ^* and thus different values of λ; one for climb and the other for descent. Denote by λ_i and λ_f the cruise cost for climb and descent, respectively; then equation (11) can be rewritten as

$$I_{up} = \underset{\substack{V_{up} \\ \pi_{up}}}{\mathrm{Min}} \frac{P - \lambda_i(V_{up} + V_{wup})}{\dot{E}]_{\dot{E}>0}}, \qquad I_{dn} = \underset{\substack{V_{dn} \\ \pi_{dn}}}{\mathrm{Min}} \frac{P - \lambda_f(V_{dn} + V_{wdn})}{|\dot{E}|_{\dot{E}<0}} \tag{11a}$$

and equation (27a) can be rewritten as

$$\left.\begin{aligned} \lambda_i &= \lambda_{ci}^*(W_{ci})(1 + \%_\lambda/100) \\ \lambda_f &= \lambda_{cf}^*(W_{cf})(1 + \%_\lambda/100) \end{aligned}\right\} \tag{27b}$$

The subscripts ci and cf refer to initial cruise and final cruise, respectively. It is clear from equations (11a) and (27b) that the initial and final cruise weights must be known before one can proceed with the climb and descent optimization.

The denominator of the Hamiltonian function is $\dot{E}$. In order to compute $\dot{E}$ (see eq. (5)), the aircraft weight must be known. The initial aircraft weight (approximately equal to the takeoff weight) is assumed to be specified;

it is the aircraft weight at energy level E_i. But the final aircraft weight (approximately equal to the landing weight) W_f, which is the aircraft weight at energy level E_f, must be estimated. The initial cruise weight, the final cruise weight, and the final aircraft weight are estimated as follows:

The initial cruise weight W_{ci} is the difference between the initial aircraft weight W_i (a known quantity) and the climb fuel F_{up}. The climb fuel is estimated by the following procedure. First, use the initial aircraft weight W_i to start the iteration, i.e., set $W_{ci} = W_i$. This assumes that the initial guess for the climb fuel is zero. Second, obtain the value of $\lambda^*(W_{ci})$ by interpreting from the cruise table the value of λ^* corresponding to W_{ci}, and compute the value of λ by the use of equation (27b) (note that $\%_\lambda$ is known here). Third, obtain the value of E_{ci}, the cruise energy, by interpolating from the cruise table the value of E corresponding to λ and use the following relation to compute the climb fuel:

$$F_{up} = K_1(E_{ci} - E_i)(1 + K_2 t_c/f_c)W_{ci}/W_{ref} \tag{28}$$

where K_1, K_2 are constants shown in table 2, E_i is the initial aircraft energy (which is computed from known initial aircraft altitude and speed), and W_{ref} is some reference initial weight, say 136,000 lb, for which the constants K_1 and K_2 were derived. The values of K_1 and K_2 shown in table 2 are for the example optimal trajection presented in this report and are obtained from curve fitting a large number of trajectories synthesized for the Boeing 727 aircraft. Fourth, test the climb fuel computed in the last step with the climb fuel computed in the previous iteration. If it is within some tolerable error, say 100 lb, then the estimation is finished. If not, update the value of the cruise weight $W_{ci} = W_i - F_{up}$ and return to the second step.

TABLE 2.- CONSTANTS FOR CLIMB-FUEL ESTIMATION

Constants	Constrained thrust	Free thrust
K_1	0.11	0.11
K_2	4.7×10^{-6}	6.1×10^{-6}

The final cruise weight W_{cf} is the difference between the initial cruise weight and F_c, the fuel consumption during cruise (cruise fuel for short). The cruise fuel is computed by

$$F_c = \dot{f}(\bar{\lambda})d_c/V_g(\bar{\lambda}) \tag{29}$$

where $\dot{f}$ is the fuel-flow rate, d_c the cruise distance, and V_g the ground speed. Both $\dot{f}$ and V_g depend on the average cruise cost $\bar{\lambda}$, which in turn is a function of aircraft weight. Procedures for obtaining the average cruise cost, the cruise fuel, final cruise weight, groundspeed, cruise distance, and final aircraft weight will now be discussed in that order.

The average cruise cost is computed in two steps: (1) determine $\lambda_c(W_c)$, the cruise cost based on the initial cruise weight, use this cruise cost to

compute the fuel-flow rate $\dot{f}(\bar{\lambda})$, and then substitute $\dot{f}(\lambda_c)$ into equation (29) to determine the approximate cruise fuel $\tilde{F}_c$; (2) estimate the average cruise weight by $\bar{W}_c = W_{ci} - \tilde{F}_c/2$ and determine the value of the optimal cruise cost $\lambda^*(\bar{W}_c)$ corresponding to $\bar{W}_c$ (from cruise table look-up, e.g., see table 3) and substitute this value into equation (27) to compute $\bar{\lambda}$. This cruise cost is used to compute the cruise fuel from the cruise table.

Strictly speaking, the cruise fuel should have been computed by integrating the fuel burn along the cruise segment of the trajectory, $F = \int_0^{d_c} [\dot{f}(x)/V_g(x)]dx$, where $\dot{f}$ and V_g are continuously updated to account for the instantaneous fuel burn along the trajectory. An example run of a 1000-n. mi. range trajectory shows that the difference in fuel burn obtained by the method stated and by integration amounts to only 1.33%. Therefore, the approximate method is considered sufficiently accurate to arrive at the optimal descent profile, since fuel burn is a small percentage of the weight of the aircraft.

The final cruise weight W_{cf} is obtained by subtracting the cruise fuel from the initial cruise weight W_{ci}. The groundspeed is computed by the use of the energy equation with the wind component taken into consideration. Specifically,

$$V_g = \sqrt{2g(E - h)} - V_w(h) \tag{30}$$

Here, E and h are obtained from searching the cruise table for known $\bar{\lambda}$, and $V_w(h)$ is obtained from the wind profile for known altitude.

The cruise distance d_c is computed by use of equation (22) for the climb-cruise-descent trajectory, and by $d_c = R - (d_{up} + d_{dn})$ for the climb-cruise*-descent trajectory. Since I_{dn} and d_{dn} are unknown quantities prior to the descent profile synthesis, d_c is also an unknown. To overcome this difficulty, a trial descent profile is generated from a rough estimate of final cruise weight and of final aircraft weight.

As a prerequisite to the trial descent profile synthesis, the cruise fuel is initially estimated by a polynomial fit in $\%_\lambda$. Similarly, the descent fuel is also estimated but by a quadratic in $\%_\lambda$. After the trial descent profile is synthesized, the cruise fuel is computed by the use of equation (2a); the descent fuel is known with sufficient accuracy.

Exact knowledge of descent fuel burn is not critical to the optimization, since the weight of the descent fuel is a small percentage of the weight of the aircraft. In fact, in some applications the weight change during descent may be considered negligible. The aircraft weight is obtained by subtracting the descent fuel from the final cruise weight. After this is done, the final cruise weight and the final aircraft weight are known fairly accurately. Then a second or refined descent profile is generated to complete the trajectory synthesis procedure.

TABLE 3.- CRUISE TABLE FOR 150,000 LB CRUISE WEIGHT

Fuel flow rate from Pratt Whitney curves
No wind run
Fuel cost, $/lb = 0.0626; time cost, $/hr = 500.00; temperature variation, K = 0.00
Aircraft cruise weight = 150000 lb

			Minimum cost/distance					Minimum fuel/time					
Altitude, ft	Min drag speed, KIAS	Max speed KIAS	KIAS	Speed, KTAS	Mach	Power setting, EPR	Fuel $/n. mi.	KIAS	Speed, KTAS	Mach	Power setting, EPR	Fuel, lb/hr	E, ft
0	239	477	391	376	0.569	1.346	3.61	225	222	0.335	1.214	8294	6254
1000	238	476	386	376	.571	1.352	3.55	225	225	.342	1.223	8204	7269
2000	241	473	381	377	.574	1.358	3.48	224	227	.346	1.233	8120	8290
3000	241	474	382	383	.585	1.372	3.42	224	231	.353	1.243	8037	9484
4000	240	472	381	387	.593	1.385	3.36	224	234	.359	1.253	7957	10618
5000	239	470	390	401	.616	1.412	3.29	224	237	.365	1.265	7881	12099
6000	242	469	390	406	.627	1.427	3.23	223	240	.370	1.277	7807	13296
7000	240	466	388	410	.635	1.439	3.17	224	245	.379	1.289	7737	14427
8000	239	463	386	413	.643	1.451	3.11	226	250	.389	1.301	7673	15554
9000	241	461	385	418	.653	1.465	3.06	224	252	.393	1.317	7622	16738
10000	240	458	384	423	.663	1.481	3.00	229	261	.409	1.329	7565	17932
11000	242	457	383	428	.673	1.495	2.95	229	265	.417	1.343	7504	19090
12000	240	454	383	435	.686	1.513	2.89	230	270	.426	1.359	7444	20353
13000	242	448	388	445	.705	1.537	2.84	229	273	.433	1.376	7386	21765
14000	243	443	384	448	.712	1.548	2.79	230	279	.443	1.392	7330	22865
15000	241	437	385	454	.726	1.566	2.73	228	281	.448	1.411	7269	24137
16000	242	431	381	457	.732	1.578	2.68	229	286	.458	1.428	7210	25226
17000	243	426	380	462	.744	1.596	2.64	228	289	.466	1.449	7156	26462
18000	244	420	376	464	.750	1.608	2.59	227	293	.473	1.470	7107	27541
19000	245	416	382	478	.776	1.638	2.54	228	298	.483	1.491	7065	29119
20000	245	410	379	481	.784	1.652	2.50	228	303	.493	1.516	7043	30255
21000	245	404	379	488	.797	1.672	2.45	232	313	.512	1.536	7024	31531
22000	246	397	372	487	.800	1.683	2.41	232	318	.522	1.562	6999	32505
23000	246	391	364	485	.799	1.693	2.37	233	324	.534	1.587	6970	33406
24000	245	385	356	483	.799	1.705	2.34	232	329	.544	1.613	6931	34309
25000	245	378	348	481	.799	1.718	2.30	232	333	.553	1.640	6890	35234
26000	247	368	341	480	.800	1.734	2.28	231	338	.564	1.669	6873	36174
27000	247	360	338	483	.809	1.758	2.25	230	342	.573	1.700	6868	37306
28000	246	351	336	487	.820	1.786	2.23	233	351	.591	1.729	6879	38502
29000	248	342	328	485	.819	1.803	2.21	238	364	.614	1.752	6890	39396
30000	249	331	321	483	.820	1.826	2.19	238	371	.629	1.784	6902	40340
31000	248	320	314	482	.821	1.852	2.17	239	378	.645	1.816	6923	41256
32000	249	308	307	479	.820	1.880	2.16	237	382	.653	1.856	6951	42150
33000	250	285	284	456	.783	1.898	2.19	238	389	.670	1.892	6995	42187
34000	245	265	265	435	.751	1.916	2.23	238	395	.682	1.933	7053	42358
35000	239	262	262	438	.759	1.963	2.22	239	403	.700	1.972	7122	43472
35500	236	259	259	437	.759	1.988	2.23	236	403	.700	2.001	7169	43937
35563	236	259	258	437	.759	1.992	2.23	236	402	.700	2.005	7175	43994
35578	236	258	258	437	.759	1.993	2.23	236	402	.700	2.006	7176	44009
35586	236	258	258	437	.759	1.993	2.23	236	402	.700	2.006	7177	44016
35588	236	258	258	437	.759	1.993	2.23	236	402	.700	2.006	7177	44018

Minimizing fuel/distance:
Optimum altitude = 32750 ft; optimum speed = 0.8198 Mach, 302 KIAS, 477 KTAS; minimum (FDOT/V) = 2.162 $/n. mi.; cruise power setting = 1.9063 EPR; optimum cruise energy = 42830 ft

Minimizing fuel/time:
Optimum altitude = 26646 ft; optimum speed = 0.5691 Mach, 230 KIAS, 340 KTAS; minimum (FDOT) = 6868 lb/hr; cruise power setting = 1.6894 EPR; optimum cruise energy = 36942 ft

SYNTHESIZING FIXED-RANGE TRAJECTORIES

A fixed-range trajectory is synthesized by matching d_t — the total ground distance (the sum of the climb profile (the cruise segment, if there is one) and the descent profile) — to the specified range R. As shown in figure 2, different values of cruise cost λ result in different ground distances. The value of λ can be specified by $\%_\lambda$, as stated in equation (23). The algorithm for synthesizing a fixed-range trajectory consists of iterating on $\%_\lambda$ (thus changing the value of d_t) if necessary until the total ground distance is within some small distance, say 5 n. mi., of R. Iteration on $\%_\lambda$ is required only if the desired range is less than some R*, the range corresponding to λ* (see table 1).

Overall Algorithm

The overall algorithm or "outer loop" consists of iteration on $\%_\lambda$ until the desired range is achieved. The larger the value of $\%_\lambda$, the shorter is the resulting distance d_t. Conversely, the smaller the value of $\%_\lambda$, the longer is the resulting distance d_t. The largest value of $\%_\lambda$ used is chosen as the lesser of 50% and the value of $\%_\lambda$ corresponding to a cruise altitude of 10,000 ft. Example calculations show that a value of 50% generates a range of about 60 n. mi. This is about the shortest range of practical interest. The smallest value of $\%_\lambda$ is 0, which corresponds to cruising at optimal cruise altitude. These extreme values of $\%_\lambda$, referred to as $\%_{\lambda_{max}}$ and $\%_{\lambda_{min}}$, establish the minimum range (ground distance d_t) R_{min} and the quantity R*, respectively, for the iteration.

Each iteration results in the synthesis of a complete trajectory. The approximately known shape of the $\%_\lambda$ vs d_t curve (see fig. 3) is used to minimize the number of iterations. To take advantage of the local shape of the curve, different regions of the curve are approximated by different functions (see PCCOMP in the Appendix). Using this procedure experience shows that about two iterations are required to reach the desired range (within 5 n. mi.).

Quantities that specify the desired trajectory include the fuel cost (f_c), time cost (t_c), temperature variation from the standard atmosphere ($\Delta\tau$), aircraft heading with respect to the wind (ψ), aircraft takeoff weight (W_i), the desired range (R), the initial altitude (h_i), the initial airspeed (V_i), the final altitude (h_f), and the final airspeed (V_f). If a cruise table is to be generated, then the maximum weight, the minimum weight and the weight interval must be specified. Other input data required include engine data, lift and drag characteristics, and wind profile.

Figure 7 shows a macroflowchart for synthesizing a fixed range optimal trajectory. It is done in the following steps (the step numbers correspond to the numbers in the blocks of the flow chart:

1. Input the engine data, lift and drag characteristics, and wind profile.

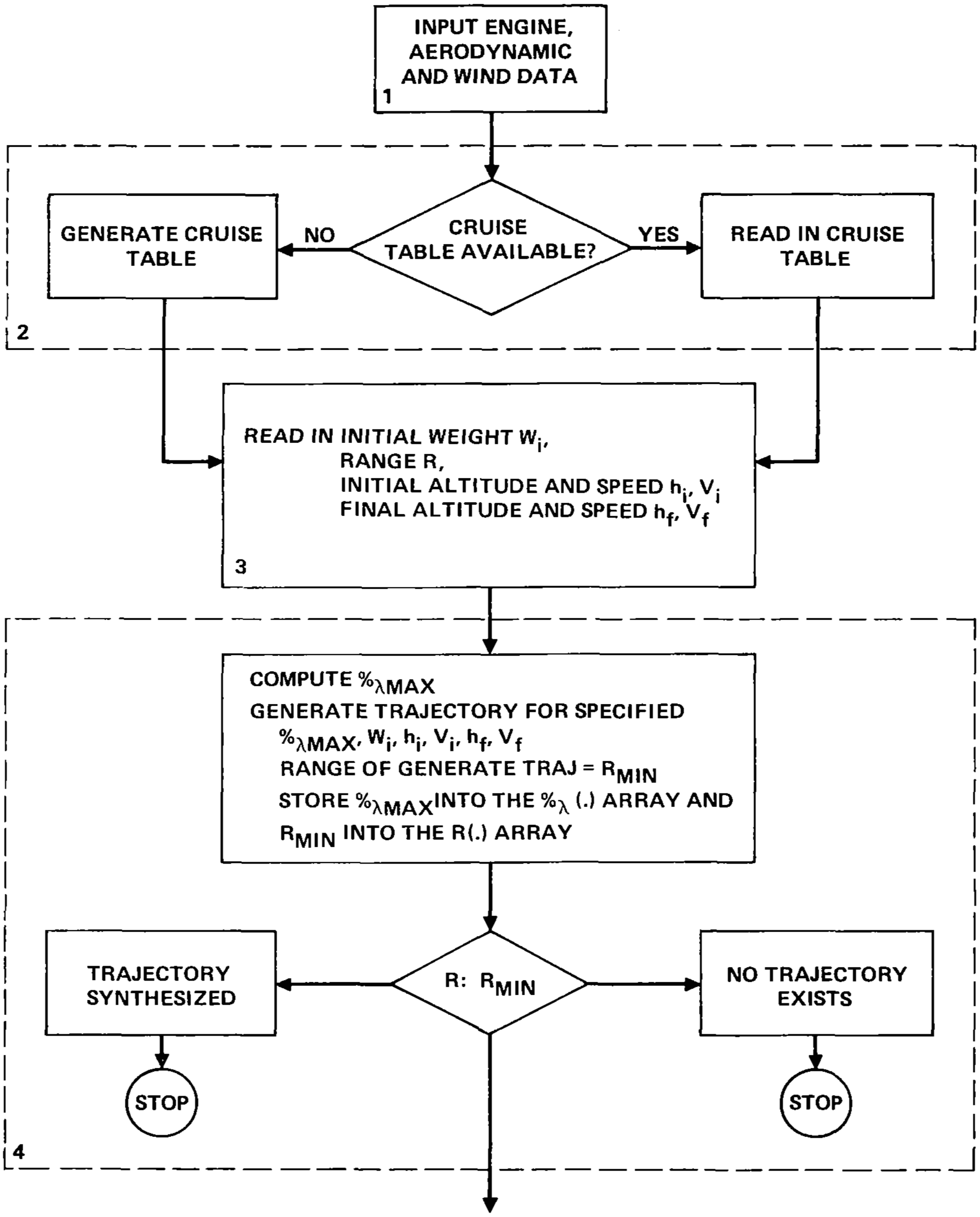


Figure 7.- Flowchart for synthesizing fixed-range optimal trajectory.

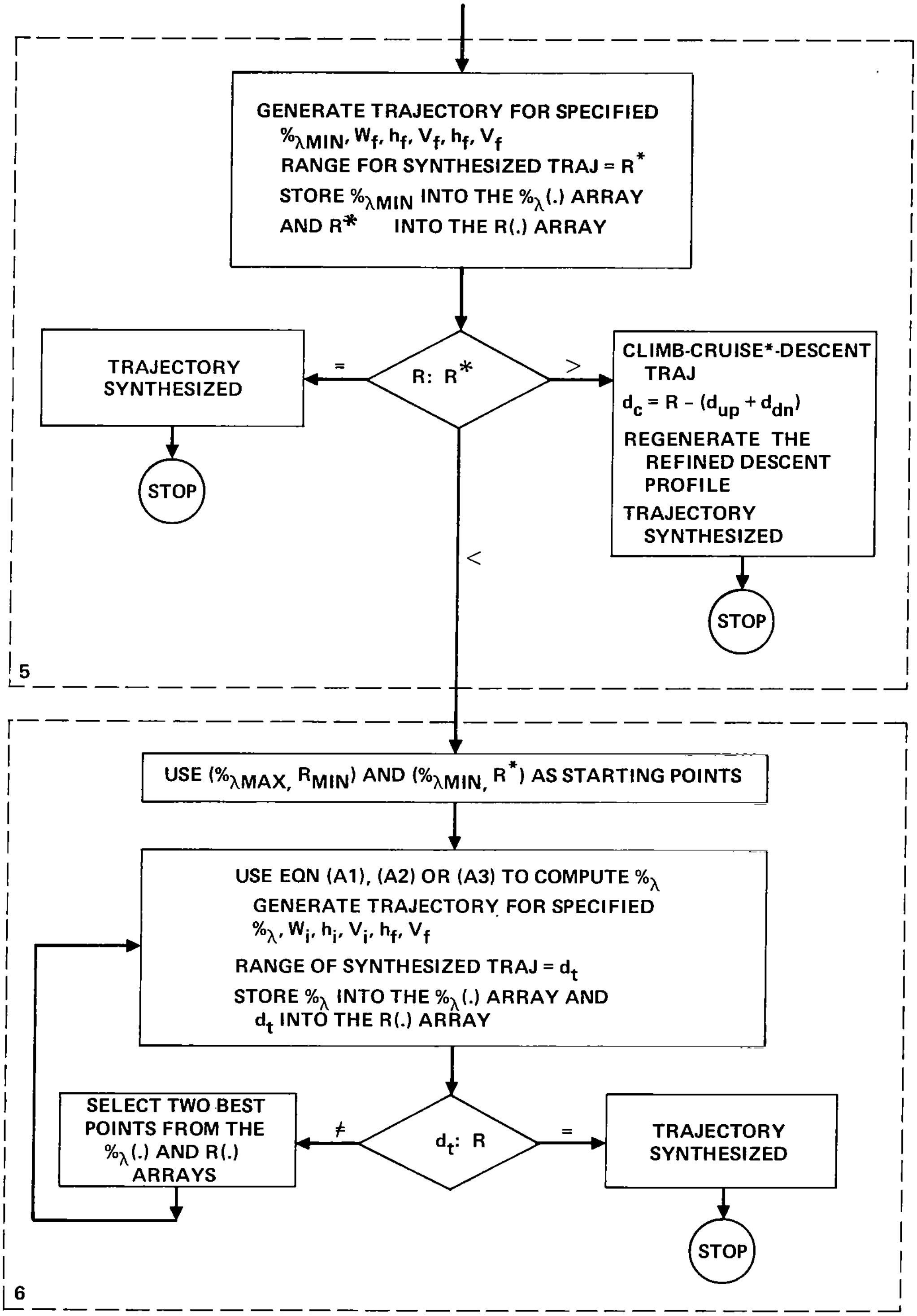


Figure 7.- Concluded.

2. Read in the cruise table if it is available and generate the cruise table if it is not.

3. Read in W_i, R, h_i, V_i, h_f, and V_f.

4. Compute $\%_{\lambda,max}$. The quantities read-in in step (3) and $\%_{\lambda,max}$ are used for generating a trajectory of range R_{min}. The desired range R is then compared with R_{min}. If R is less than R_{min}, then no trajectory is computed. If it is within ε of R_{min}, then a trajectory has been synthesized. If greater, then proceed.

5. In a manner similar to that of step (4), use $\%_{\lambda_{min}}$ and the known quantities from step (3) to generate a trajectory of range R* and compare R with R*. If R is less than R* and is within ε distance of R*, then a trajectory has been synthesized. If R is greater than R*, then the trajectory is a climb-cruise*-descent type. In this case, the cruise distance is computed from $d_c = R - (d_{up} + d_{dn})$. The final weight is then determined. Next, this updated cruise weight is used in regenerating the refined descent profile, thus completing the climb-cruise*-descent trajectory.

6. Use $(\%_{\lambda_{max}}, R_{min})$ and $(\%_{\lambda_{min}}, R_{max})$ as the starting points to compute a new value of $\%_\lambda$ and iterate on $\%_\lambda$ until the desired range trajectory is synthesized. The solution always exists because the fact that $\%_\lambda$ is within $(\%_{\lambda_{min}}, \%_{\lambda_{max}})$ which implies R is within (R_{min}, R_{max}).

Generating a Trajectory for Fixed Value of $\%_\lambda$

Generating a trajectory for fixed value of $\%_\lambda$ consists primarily of generating the climb and descent profiles and estimating the final cruise weight and the landing weight.

Figure 8 is a macroflowchart showing these steps in detail. The known quantities are $\%_\lambda$, E_i, E_f (the initial and final aircraft energies, respectively) and W_i, the aircraft takeoff weight (strictly speaking, the aircraft weight at the beginning of climb). The energies E_i and E_f are computed from known initial aircraft altitude and speed and from specified final aircraft altitude and speed, respectively. The trajectory is generated in the following steps:

1. Estimate the initial cruise weight W_{ci} and use it in conjunction with $\%_\lambda$ to obtain $\lambda_i(W_{ci})$ and E_{ci}, the initial cruise cost and cruise energy, respectively. The initial cruise weight is obtained by subtracting the climb fuel F_{up} from takeoff weight W_i. The climb fuel is estimated as described in the preceding section, and equation (29) is used to compute the climb fuel. The optimal cruise cost $\lambda^*(W_{ci})$ corresponding to W_{ci} is obtained from the cruise table (see, e.g., table 3). Next, the cruise cost for climb optimation $\lambda_i(W_{ci})$ is computed by the use of equation (28). The cruise energy E_{ci} is obtained from the cruise table for the corresponding value of cruise cost $\lambda_i(W_{ci})$.

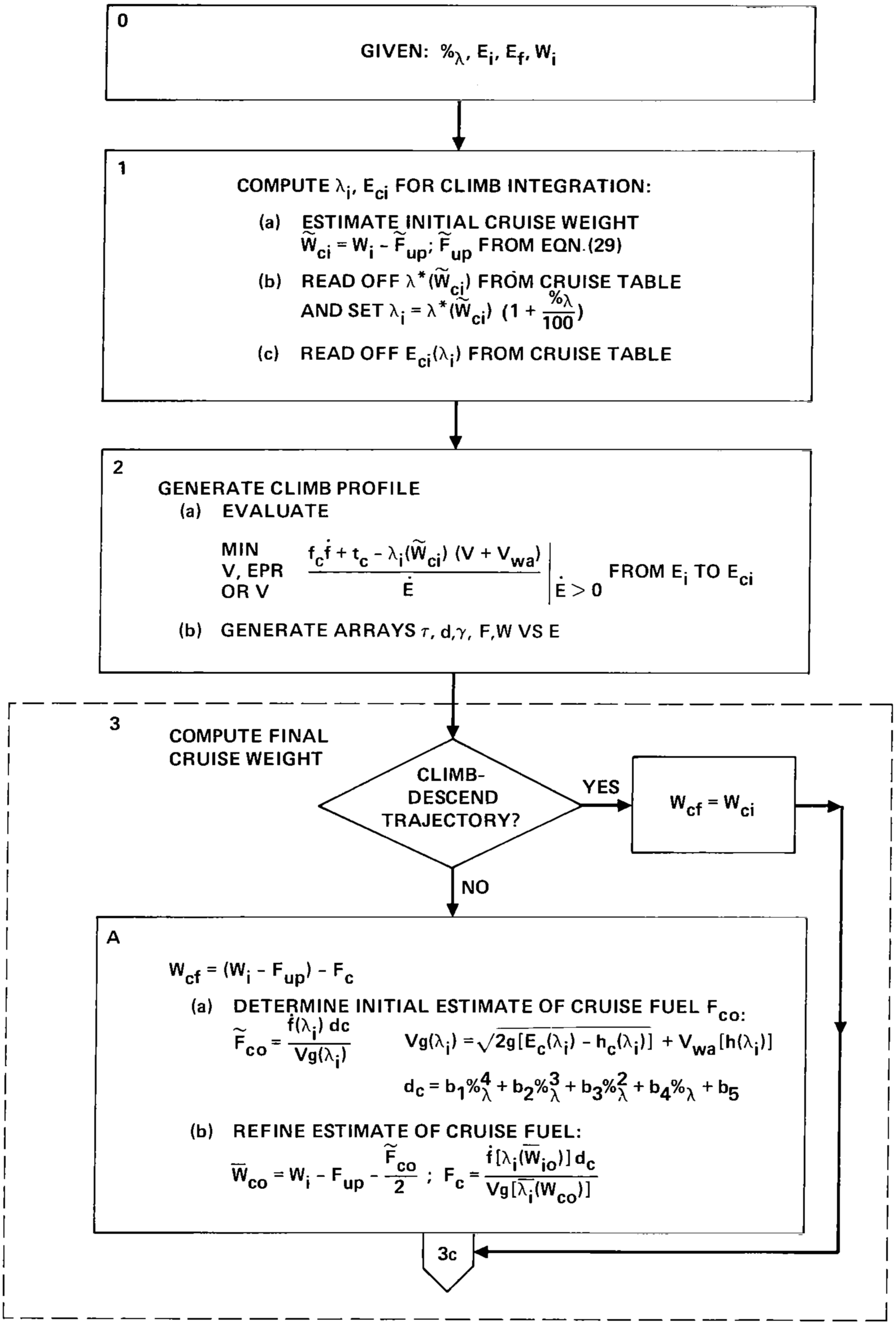


Figure 8.- Flowchart for generating trajectory with fixed $\%_\lambda$.

3c

4

ESTIMATE λ_f, E_{cf} FOR DESCENT INTEGRATION:

(a) READ OFF $\lambda^*(W_{cf})$ FROM CRUISE TABLE

$\lambda_f = \lambda^*(W_{cf})\ (1 + \%\lambda/100)$

(b) READ OFF $E_{cf}(\lambda_f)$ FROM CRUISE TABLE

↓

5

ESTIMATE LANDING WEIGHT W_f:

$\tilde{W}_{fo} = W_{cf} - \tilde{F}_{dn}$

$\tilde{F}_{dn} = b_1 \%_\lambda^2 + b_2 \%_\lambda + b_3$

↓

6

GENERATE TRIAL DESCENT PROFILE; EVALUATE:

$$\min_{V,EPR \text{ OR } V} \left. \frac{f_c \dot{f} + t_c - \lambda_f(W_{cf})\,(V + V_{wa})}{|\dot{E}|} \right|_{\dot{E} < 0} \quad \text{FROM } E_f \text{ TO } E_{cf}$$

↓

7

REFINE FINAL CRUISE AND LANDING WT COMPUTATION:

$$W_{cf} = W_i - F_{up} - F_c \text{ (SEE STEP 3)}; \quad d_c = \left. \frac{I_{up} + I_{dn}}{-\frac{d\lambda}{dE}} \right|_{E_i(\lambda_i)}$$

$$W_f = W_i - (F_{up} + F_c + F_{dn})$$

↓

8

COMPUTE EXACT λ_f AND E_{cf}; GENERATE REFINED DESCENT PROFILE:

(a) EVALUATE

$$\min_{V,\,EPR \text{ OR } V} \left. \frac{f_c \dot{f} + t_c - \lambda_f(W_{cf})\,(V + V_{wa})}{|\dot{E}|} \right|_{\dot{E} < 0} \quad \text{FROM } E_f \text{ TO } E_{cf}$$

(b) GENERATE ARRAYS τ, d, γ, F,W VS E

(c) COMPUTE PERTINENT QUANTITIES WHICH SUMMARIZE THE GENERATED TRAJECTORY.

Figure 8.- Concluded.

2. Generate the climb profile by minimizing the Hamiltonian function (eq. (11)) for climb, starting from E_i and ending at E_{ci}. The cruise cost obtained in step (1) is used in evaluation of the Hamiltonian function. Also, generate time histories of the distance traversed d, the flightpath angle γ, the fuel burn F, and the aircraft weight W, by use of equations (16) through (19).

3. Estimate the final cruise weight W_{cf} and the landing weight W_f. The final cruise weight is obtained by subtracting the cruise fuel from the initial cruise weight. For climb-descent type trajectories the cruise fuel is zero and W_{cf} is obtained by equating it to W_{ci}. For any other type of trajectory, it is computed by the use of equation (29). The numerator of equation (29) contains d_c, the cruise distance. This distance can be computed by the use of equation (22). The numerator of (22) contains I_{dn}, an unknown quantity prior to the synthesis of a descent profile. To overcome this difficulty, the cruise distance is estimated by a polynomial in $\%_\lambda$.

4. In a manner similar to that of step (1), compute the cruise cost for descent λ_f (W_{cf}) and the final cruise energy E_{cf} by reference to the cruise table and by substituting the value of $\%_\lambda$ into equation (27b). For the climb-descent type of trajectory, they are obtained by equating them to corresponding initial cruise cost and initial cruise energy.

5. Obtain the final weight by subtracting the descent fuel from the final cruise weight. The descent fuel F_{dn} is estimated by a quadratic in $\%_\lambda$.

6. In a manner similar to that of step (2) generate the trial descent profile by minimizing the Hamiltonian function for descent, starting from E_f to E_{cf}, and generate a time history of d, γ, F, and W.

7. Compute the exact final cruise weight and exact final weight. Since a descent profile is now available, I_{dn} is known and d_c can now be computed exactly. Similarly, since both F_{dn} and W_{cf} are also known, W_f can also be computed exactly.

8. Compute λ_f and E_{cf} based on the refined value of W_{cf}, generate the refined descent profile, and compute pertinent quantities for all segments of the generated trajectory.

EXAMPLE COMPUTER PROGRAM OUTPUT

The computer program is intended to provide a wide range of users with useful information about the trajectory being synthesized. The information output by the computer was evolved after a series of consultations with aircraft manufacturers, commercial airline personnel, and other users. Output information includes (1) optimum cruise tables at selected weights, (2) a summary of optimal cruise quantities as a function of cruise weight, (3) the optimal cruise quantities as a function of cruise distance, (4) $d\lambda/dE$ for selected cruise weights, (5) climb profile, (6) descent profile, and (7) a summary of the complete synthesized trajectory. In this section, the computer

program output will be illustrated by considering a sample case for a model of the Boeing 727 aircraft.

Computation of the cruise table is the first step. This table contains all precomputed quantities for synthesizing the cruise segment of known cruise distance. It also provides the cruise cost for generating the climb and descent profiles. Each cruise table is computed for specified values of time cost, fuel cost, wind profile, and atmospheric condition. The table is divided into pages, with each page containing the pertinent quantities for a fixed cruise weight.

A typical page of the cruise table for the example aircraft is shown in table 3. The parameters that define the case are $0.0626/lb fuel cost, $500/hr time cost, zero temperature variation from the standard atmospheric condition, and zero wind. The page shown is for a 150,000 lb cruise weight. The first column is the altitude in feet, incremented at 1,000-ft intervals, except in the vicinity of the ceiling, where the increment is decreased. Columns (2) and (3) are the corresponding minimum drag speed and the maximum allowable speed, both in KIAS (indicated airspeed in knots). Columns (4) through (8) are the results of minimization of equation (12) for the given altitude. Columns (4), (5), and (6) are the optimal cruise speed in KIAS, KTAS (true airspeed in knots), and Mach number, respectively. Column (7) is the corresponding power setting expressed in engine pressure ratio (EPR); column (8) is the optimal cruise cost in $/n. mi. Similarly, columns (9) through (13) are the results of minimization of the fuel flow at each altitude. This information is not used in the optimum trajectory synthesis, but is of interest since it specifies the optimum holding speed. Columns (9), (10), and (11) are the optimal holding speeds in KIAS, KTAS, and Mach number. Columns (12) and (13) are the EPR setting and fuel-flow rate in lb/hr, which is the minimum fuel-flow rate achievable at that altitude. Column (14) is the cruise energy in feet. The last entry in the cruise table (i.e., an altitude of 35,588 ft) gives conditions at the ceiling. Additional information at the bottom of the page gives the optimal cruise quantities for the specified cruise weight, i.e., 150,000 lb. They are obtained as a result of minimizing the cost function, expressed by equation (21b), with respect to both altitude h and airspeed V. The optimal cruise altitude for the example case is shown to be 32,750 ft. The corresponding optimal cruise speeds are 302 KIAS, 477 KTAS, or 0.8198 Mach number. The optimal cruise cost is $2.162/n. mi., optimal EPR setting, 1.9069, and optimal cruise energy, 42,830 ft.

A similar set of quantities is obtained by minimizing the fuel-flow rate with respect to both h and V. As shown at the bottom of the page, the optimal cruise altitude for this case is 26,646 ft. The optimal cruise speed is 230 KIAS, 340 KTAS, and 0.5691 Mach number. The minimum fuel flow is 6,868 lb/hr, the cruise-energy is 36,942 ft, and the cruise-power setting is 1.6906 EPR.

Table 4 is a summary of the optimal quantities as a function of aircraft weight at cruise. Column (1) is the cruise weight. Column (2) is the corresponding optimal cruise altitude; it increases as the cruise weight decreases. Columns (3) and (4) show the optimal cruise speed in KTAS and Mach number, respectively, and column (5) gives the EPR setting. Column (6) shows the

TABLE 4.- OPTIMAL CRUISE ENERGY, SPEED, AND ALTITUDE VS WEIGHT

Cruise wt, lb	Opt H, ft	KTAS	Opt Mach	EPR setting	Cost, \$/n. mi.	Fuel flow, lb/hr	Opt E, ft
150000	32760	477.35	0.8198	1.9069	2.162	8490.71	42842
145000	33442	475.90	.8198	1.9052	2.130	8194.45	43463
140000	34176	474.35	.8198	1.9050	2.097	7898.65	44132
135000	34918	472.77	.8198	1.9039	2.065	7602.05	44808
130000	35721	471.05	.8198	1.9042	2.033	7302.05	45540
125000	37065	469.93	.8194	1.9248	1.998	7006.03	46836
120000	37911	469.93	.8194	1.9245	1.963	6739.86	47683
115000	38792	469.93	.8194	1.9244	1.927	6474.53	48564
110000	39689	469.93	.8194	1.9231	1.892	6207.10	49460

cruise cost in \$/n. mi.; it decreases with increasing cruise weight. Column (7) shows the minimum fuel-flow rate which decreases with decreasing cruise weight. Finally, column (8) shows the optimal cruise energy. Although the outputs are for a specific aircraft, these trends are generally true for all subsonic transport aircraft.

Table 5 shows the optimal cruise quantities as a function of cruise distance. This table assumes the initial cruise weight is known. As the aircraft cruises, it loses weight because of fuel burn. Since the optimal cruise quantities depend on cruise weight, their values change as the aircraft loses weight. This table contains the new quantities to account for the change in weight, in this case at increments of 100 n. mi. cruising distance. Column (1) shows the cruise distance in nautical miles, column (2) shows the corresponding time from the start of cruise in hr:min:sec, and column (3) shows the cruise weight updated to account for fuel burn. Columns (4) and (5) give the cruise energy and the cruise altitude, respectively, both in feet. Columns (6) and (7) show the cruise speed in Mach number and KIAS, respectively, and column (8) gives the ground speed in knots. Column (9) shows the cruise cost in dollars per nautical mile, and column (10) gives the power setting in engine pressure ratio.

Table 6 displays the coefficients of a polynomial for computing the derivative $d\lambda/dE$, where $d\lambda/dE = AE + B$ and the values of A and B depend on cruise weight (see fig. 1). Column (1) of the table shows the cruise weight in pounds, column (2) the value of A, and column (3) the value of B.

Table 7 shows the output for a climb optimization for unconstrained thrust. The aircraft takeoff weight is 150,000 lb and the cruise weight is estimated to be 145,356 lb. The cruise energy (energy height for terminating the climb integration) is 37,468 energy feet. The initial altitude and speed are 0 ft and 210 KIAS, respectively. As before the fuel cost is \$0.0626/lb and the time cost \$500/hr. The temperature variation from the standard profile is zero. The cruise cost corresponding to the estimated cruise weight is \$2.231/n. mi. For computational numerical stability, it is set to a value

TABLE 5.- CRUISE QUANTITIES AS FUNCTION OF CRUISE DISTANCE

Cruise dist, n. mi.	Time hr:mn:sec	Weight, lb	Energy, ft	Altitude, ft	Mach no.	KIAS	Ground speed, knot	λ, \$/n. mi.	Power setting EPR
0.00	0: 0: 0	144874	43480	33460	0.8198	475.87	475.87	2.129	1.905
100.00	0:12:37	143181	43707	33709	.8198	475.34	475.34	2.118	1.905
200.00	0:25:15	141488	43933	33958	.8198	474.81	474.81	2.107	1.905
300.00	0:37:53	139802	44159	34206	.8198	474.28	474.28	2.096	1.905
400.00	0:50:33	138166	44380	34448	.8198	473.77	473.77	2.086	1.905
500.00	1: 3:13	136529	44601	34691	.8198	473.25	473.25	2.075	1.904
600.00	1:15:54	134896	44823	34934	.8198	472.73	472.73	2.065	1.904
700.00	1:28:37	133317	45054	35188	.8198	472.19	472.19	2.054	1.904
800.00	1:41:19	131738	45285	35442	.8198	471.65	471.65	2.044	1.904
900.00	1:54: 2	130159	45516	35696	.8198	471.11	471.11	2.034	1.904
1000.00	2: 6:47	128632	45894	36089	.8197	470.75	470.75	2.023	1.910

TABLE 6.- COEFFICIENTS FOR dλ/dE FOR DIFFERENT CRUISE WEIGHT

dλ/dE = A E + B in \$/n. mi.**2

Cruise wt	A	B
150000	0.1838019E-04	-0.8058240E 00
145000	.1831762E-04	-.8144628E 00
140000	.1645151E-04	-.7424970E 00
135000	.1588525E-04	-.7276682E 00
130000	.1502870E-04	-.6994290E 00
125000	.1498086E-04	-.7166300E 00
120000	.1357229E-04	-.6607372E 00
115000	.1190378E-04	-.5899974E 00
110000	.1106663E-04	-.5584263E 00

TABLE 7.- CLIMB PROFILE

Aircraft takeoff weight = 150000 lb; initial weight = 145356 lb; cruise energy = 37468 ft
Initial altitude, ft; speed, KIAS = 0   210
Fuel cost, $/lb = 0.0626; time cost, $/hr = 500.00; temperature variation, K = 0.00; lambda = 2.231 $/n. mi.
Climb optimization:

Energy, ft	Altitude, ft	Mach	VIAS, knots	VTAS, knots	EDOT, ft/s	Gamma, deg	Time, hr:min:s	Dist., n. mi.	Fuel used, lb	Power setting, EPR	Cost/E, $/E ft
1951	1	0.317	213	210	49.22	0.00	0: 0:10	0.000	61	1.802	47.213
2069	4	.327	219	216	50.93	.04	0: 0:19	0.581	119	1.800	45.289
2569	11	.364	244	240	56.20	.11	0: 0:28	1.144	173	1.791	39.824
3069	309	.378	253	250	57.16	4.73	0: 0:37	1.737	226	1.791	38.400
3569	785	.380	252	251	56.50	7.31	0: 0:46	2.347	278	1.796	38.287
4069	1246	.384	253	253	55.96	6.98	0: 0:55	2.967	331	1.801	38.082
4569	1708	.387	253	254	55.39	6.87	0: 1: 4	3.598	383	1.806	37.893
5069	2169	.390	253	256	54.85	6.74	0: 1:13	4.239	436	1.811	37.699
5569	2629	.393	253	258	54.31	6.62	0: 1:22	4.891	488	1.817	37.506
6069	3089	.397	253	260	53.77	6.50	0: 1:32	5.555	540	1.822	37.315
6569	3548	.400	253	261	53.22	6.38	0: 1:41	6.230	593	1.827	37.126
7069	4006	.403	253	263	52.72	6.27	0: 1:50	6.916	645	1.832	36.904
7569	4464	.407	253	265	52.21	6.16	0: 2: 0	7.614	697	1.837	36.684
8069	4921	.410	253	267	51.69	6.05	0: 2:10	8.324	749	1.842	36.467
8569	5377	.414	253	269	51.17	5.93	0: 2:19	9.046	801	1.848	36.252
9069	5833	.417	253	270	50.65	5.82	0: 2:29	9.781	852	1.853	36.040
9569	6288	.421	253	272	50.12	5.71	0: 2:39	10.529	904	1.858	35.830
10069	6743	.425	253	274	49.58	5.61	0: 2:49	11.290	956	1.863	35.623
10569	7196	.428	254	276	48.72	5.46	0: 3: 0	12.070	1008	1.864	35.418
11069	7649	.432	254	278	48.49	5.39	0: 3:10	12.860	1060	1.873	35.215
11569	8101	.436	254	280	47.96	5.28	0: 3:20	13.664	1111	1.878	35.010
12069	8553	.439	254	282	47.42	5.18	0: 3:31	14.483	1163	1.883	34.808
12569	9003	.443	254	284	46.87	5.08	0: 3:42	15.318	1215	1.888	34.608
13069	9453	.447	254	286	46.32	4.97	0: 3:52	16.168	1267	1.893	34.411
13569	9902	.451	254	288	45.77	4.87	0: 4: 3	17.035	1319	1.898	34.215
14069	10351	.455	254	290	45.20	4.77	0: 4:14	17.919	1370	1.904	34.025
14569	10372	.483	271	308	46.48	0.23	0: 4:25	18.812	1421	1.894	32.007
15069	10398	.510	287	325	47.14	0.26	0: 4:36	19.744	1472	1.885	30.573
15569	10428	.535	301	341	47.48	0.30	0: 4:46	20.717	1523	1.875	29.448
16069	10464	.559	316	356	47.29	0.32	0: 4:57	21.739	1574	1.867	28.721
16569	10503	.581	329	370	46.65	0.35	0: 5: 8	22.820	1627	1.858	28.320
17069	10856	.589	332	375	45.57	2.93	0: 5:19	23.953	1680	1.856	28.131
17569	11309	.592	331	376	44.93	3.69	0: 5:30	25.111	1734	1.861	27.952
18069	11402	.611	342	388	44.41	0.73	0: 5:41	26.305	1789	1.858	27.733
18569	11852	.615	341	390	43.81	3.45	0: 5:52	27.535	1844	1.863	27.513
19069	12307	.618	340	391	43.21	3.42	0: 6: 4	28.787	1898	1.868	27.292
19569	12767	.621	338	392	42.60	3.39	0: 6:16	30.060	1954	1.873	27.071
20069	13226	.624	337	393	41.69	3.32	0: 6:28	31.365	2009	1.875	26.845
20569	13634	.629	337	396	41.39	2.90	0: 6:40	32.687	2064	1.882	26.618
21069	14094	.632	336	397	40.80	3.21	0: 6:52	34.034	2120	1.888	26.390
21569	14504	.637	336	400	40.17	2.81	0: 7: 4	35.409	2176	1.891	26.161
22069	14966	.640	335	401	39.27	3.08	0: 7:17	36.821	2232	1.894	25.931
22569	15375	.645	335	403	38.64	2.67	0: 7:30	38.263	2289	1.898	25.699
23069	15835	.648	333	404	38.04	2.95	0: 7:43	39.735	2346	1.903	25.466
23569	16242	.653	333	407	37.40	2.55	0: 7:57	41.239	2403	1.907	25.231
24069	16704	.656	332	408	37.08	2.86	0: 8:10	42.762	2461	1.915	24.994
24569	17113	.661	332	411	36.15	2.45	0: 8:24	44.332	2519	1.916	24.754
25069	17572	.664	331	412	35.55	2.70	0: 8:38	45.935	2577	1.921	24.510
25569	17977	.669	331	414	34.90	2.35	0: 8:52	47.576	2636	1.925	24.263
26069	18437	.672	329	415	34.55	2.60	0: 9: 7	49.241	2695	1.933	24.013
26569	18843	.677	329	418	33.89	2.24	0: 9:22	50.946	2754	1.937	23.759
27069	19249	.682	329	420	33.23	2.19	0: 9:37	52.696	2815	1.941	23.500
27569	19705	.686	328	422	32.59	2.40	0: 9:52	54.487	2875	1.946	23.238
28069	20109	.691	328	424	31.67	2.05	0:10: 8	56.340	2936	1.946	22.971
28569	20261	.706	335	433	31.04	0.75	0:10:24	58.257	3000	1.944	22.687
29069	20633	.713	336	437	30.39	1.77	0:10:40	60.243	3064	1.947	22.364
29569	21060	.717	335	439	29.78	1.97	0:10:57	62.282	3128	1.951	22.035
30069	21483	.721	334	441	29.16	1.91	0:11:14	64.372	3193	1.955	21.698
30569	21903	.726	334	442	28.55	1.87	0:11:32	66.518	3259	1.959	21.354
31069	22350	.730	333	444	27.70	1.79	0:11:50	68.739	3326	1.960	21.002
31569	22698	.737	334	448	27.04	1.51	0:12: 8	71.029	3394	1.962	20.640
32069	23117	.742	333	450	26.41	1.68	0:12:27	73.386	3463	1.966	20.268
32569	23554	.746	333	452	25.99	1.63	0:12:46	75.793	3533	1.973	19.885
33069	23952	.751	332	454	25.15	1.57	0:13: 6	78.292	3603	1.974	19.491
33569	24358	.757	332	457	24.49	1.41	0:13:27	80.873	3675	1.977	19.082
34069	24779	.761	331	458	23.87	1.56	0:13:48	83.532	3748	1.981	18.665
34569	25125	.768	332	462	22.68	2.32	0:13:59	84.939	3786	1.978	18.288
34819	25266	.773	333	465	22.77	0.94	0:14:10	86.352	3824	1.984	18.111
35069	25450	.776	334	466	22.32	1.20	0:14:21	87.799	3863	1.984	17.930
35319	25620	.780	334	468	21.87	1.08	0:14:32	89.281	3902.	1.984	17.743
35569	25808	.783	334	470	21.37	1.16	0:14:44	90.804	3942.	1.983	17.551
35819	25974	.787	335	472	20.06	0.96	0:14:56	92.432	3984.	1.972	17.349
36069	26127	.791	336	474	18.67	0.82	0:15:10	94.190	4028.	1.959	17.133
36319	26319	.794	336	475	17.00	0.93	0:15:25	96.128	4076.	1.944	16.895
36569	26474	.799	337	478	15.48	0.68	0:15:41	98.265	4128.	1.929	16.636
36819	26707	.800	336	478	13.01	0.86	0:16: 0	100.814	4186.	1.904	16.319
37069	26981	.800	334	478	9.91	0.77	0:16:25	104.161	4260.	1.870	15.905
37319	27748	.782	320	465	5.15	1.14	0:17:14	110.512	4386.	1.812	15.027
37518	27825	.787	322	468	5.01	0.11	0:18: 4	116.973	4515.	1.814	14.278

slightly higher than the optimal cruise cost, specifically, 1% above. The tabulated data show the climb integration; each line represents an energy level that starts from the initial energy E_i, increments at 500 energy feet (except near the cruise energy, where the increment is decreased), and terminates at an energy level higher than the cruise energy. (A slightly higher final energy level provides an extra point in the table. This point is used for interpolation).

In table 7, column (1) shows the energy height in feet; column (2) shows the altitude; and columns (3), (4), and (5) show the aircraft speed in Mach number, KIAS, and KTAS, respectively. Column (6) shows the rate of energy change and column (7) shows the flightpath angle in degrees. The flightpath angle as a function of energy shows discontinuities at several places. This is due to the use of tabulated engine data with a low-order dynamic model (i.e., neglecting rotational dynamics). These discontinuities could be eliminated if some smoothing or averaging process were used. Column (8) shows the time into the climb in hr:min:sec; column (9) is the distance traversed in nautical miles; and column (10) is the cumulative climb fuel in pounds. Column (11) shows the EPR setting and column (12) the minimized Hamiltonian function I_{up}.

Table 8 shows the result of integrating the descent trajectories for the same example aircraft and the unconstrained-thrust case. The landing weight is estimated to be 144,843 lb, and the cruise energy is 37,461 ft, the same as that for the climb since there is no cruise segment in this case. The final altitude and speed are 0 ft and 210 KIAS, respectively. The outputs are very similar to those for the climb optimization. Quantities shown in columns (1) through (11) correspond to those in the climb table. The energy rate, as shown in column (6), is negative. The EPR setting, shown in column (11), starts with 1.450 at cruise altitude and switches to idle at or below an altitude of 24,490 ft. Column (12) shows I_{dn}, the minimized Hamiltonian function for descent. Column (13) shows the sum of I_{up} and I_{dn}, the sum of the minimized Hamiltonian function for both climb and descent. This sum is used to evaluate the cruise distance for the constrained-thrust case (see eq. (22)). It is also used to locate the altitude at which the climb profile joins the descent profile in the unconstrained-thrust case. This altitude is the zero-intercept of $(I_{up} + I_{dn})$ vs altitude.

Table 9 summarizes the synthesized trajectory. The first part of the table gives the aircraft weight in pounds; cost in dollars per nautical mile; energy height in feet; altitude in feet; true airspeed, indicated airspeed, and groundspeed, all in knots; and the Mach number for both initial and final cruise conditions. For the climb-descend type of trajectory, the corresponding quantities for the initial and final cruise are the same, since the aircraft weight at the end of a climb is the same as that at the beginning of a descent (i.e., no cruise segment).

The second part of table 9 shows the fuel used in pounds, the ground track distance covered in nautical miles, the elapsed time in hr:min:sec, the cost in dollars, and the unit cost in dollars per nautical mile for the climb profile, descent profile, cruise segment, and synthesized overall trajectory. The last section of the table shows the final aircraft weight in pounds, and

TABLE 8.- DESCENT PROFILE

Aircraft landing weight = 144843 lb; cruise energy = 37461 ft
Final altitude, ft; speed, KIAS = 0 210
Descend optimization:

Energy, ft	Altitude, ft	Mach	VIAS, knots	VTAS, knots	EDOT, ft/s	Gamma, deg	Time, hr:min:s	Dist., n. mi.	Fuel used, lb	Power setting, EPR	Cost/E, $/E ft	Sum cost/E, $/E ft
1951	2	0.317	213	210	-21.95	0.00	0: 0:22	0.000	19	IDLE	16.578	63.791
2062	5	.326	219	216	-22.32	-0.02	0: 0:45	1.323	37	IDLE	15.388	60.678
2562	11	.363	244	240	-25.15	-0.05	0: 1: 5	2.581	54	IDLE	10.218	50.042
3062	303	.378	253	250	-26.73	-2.16	0: 1:23	3.852	69	IDLE	8.217	46.616
3562	779	.380	252	251	-26.86	-3.47	0: 1:42	5.143	84	IDLE	7.830	46.117
4062	1240	.384	253	253	-27.12	-3.37	0: 2: 0	6.429	99	IDLE	7.329	45.411
4562	1702	.387	253	254	-27.36	-3.39	0: 2:19	7.712	114	IDLE	6.844	44.736
5062	2163	.390	253	256	-27.61	-3.39	0: 2:37	8.993	128	IDLE	6.363	44.062
5562	2623	.393	253	258	-27.86	-3.39	0: 2:55	10.271	141	IDLE	5.888	43.394
6062	3083	.397	253	260	-28.11	-3.39	0: 3:12	11.546	155	IDLE	5.419	42.734
6562	3542	.400	253	261	-28.36	-3.40	0: 3:30	12.818	168	IDLE	4.955	42.081
7062	4000	.403	253	263	-28.62	-3.40	0: 3:48	14.088	180	IDLE	4.502	41.406
7562	4458	.407	253	265	-28.88	-3.40	0: 4: 5	15.354	193	IDLE	4.053	40.737
8062	4915	.410	253	267	-29.14	-3.40	0: 4:22	16.618	205	IDLE	3.610	40.077
8562	5371	.414	253	269	-29.39	-3.41	0: 4:39	17.880	216	IDLE	3.201	39.454
9062	5827	.417	253	270	-29.65	-3.41	0: 4:56	19.140	228	IDLE	2.805	38.845
9562	6282	.421	253	272	-29.90	-3.41	0: 5:13	20.397	239	IDLE	2.413	38.243
10062	6737	.425	253	274	-30.16	-3.41	0: 5:29	21.652	250	IDLE	2.026	37.649
10562	7190	.428	254	276	-30.41	-3.41	0: 5:46	22.906	261	IDLE	1.643	37.061
11062	7643	.432	254	278	-30.67	-3.41	0: 6: 2	24.158	271	IDLE	1.265	36.480
11562	8095	.436	254	280	-30.92	-3.41	0: 6:18	25.408	281	IDLE	0.891	35.902
12062	8547	.439	254	282	-31.18	-3.40	0: 6:34	26.656	291	IDLE	0.522	35.330
12562	8997	.443	254	284	-31.44	-3.40	0: 6:50	27.903	300	IDLE	0.156	34.764
13062	9447	.447	254	286	-31.69	-3.40	0: 7: 6	29.148	310	IDLE	-0.206	34.205
13562	9897	.451	254	288	-31.95	-3.40	0: 7:21	30.392	319	IDLE	-0.563	33.652
14062	10345	.455	254	290	-32.21	-3.40	0: 7:37	31.635	328	IDLE	-0.903	33.122
14562	10366	.483	271	308	-35.76	-0.17	0: 7:51	32.795	336	IDLE	-2.613	29.394
15062	10392	.510	287	325	-39.75	-0.22	0: 8: 4	33.900	344	IDLE	-3.856	26.718
15562	10423	.535	301	341	-44.08	-0.28	0: 8:15	34.948	351	IDLE	-4.739	24.708
16062	10458	.559	316	356	-48.65	-0.34	0: 8:25	35.942	357	IDLE	-5.377	23.344
16562	10499	.581	329	370	-53.41	-0.40	0: 8:35	36.885	363	IDLE	-5.836	22.484
17062	10543	.603	342	384	-58.35	-0.47	0: 8:43	37.782	368	IDLE	-6.160	21.971
17562	10593	.623	355	397	-63.49	-0.55	0: 8:51	38.636	373	IDLE	-6.369	21.583
18062	10916	.632	360	402	-65.09	-3.57	0: 8:59	39.486	378	IDLE	-6.522	21.211
18562	11468	.631	357	400	-63.64	-5.96	0: 9: 7	40.356	383	IDLE	-6.673	20.839
19062	11951	.632	354	401	-62.99	-5.27	0: 9:14	41.235	388	IDLE	-6.828	20.464
19562	12455	.634	351	401	-62.33	-5.22	0: 9:22	42.124	393	IDLE	-6.985	20.086
20062	12949	.635	349	401	-61.68	-5.17	0: 9:31	43.023	398	IDLE	-7.145	19.700
20562	13498	.634	345	400	-60.37	-5.63	0: 9:39	43.939	403	IDLE	-7.308	19.310
21062	13988	.636	342	400	-59.80	-4.99	0: 9:47	44.863	407	IDLE	-7.473	18.917
21562	14479	.638	340	400	-59.22	-4.94	0: 9:56	45.797	412	IDLE	-7.641	18.520
22062	14971	.639	337	400	-58.63	-4.90	0:10: 4	46.741	417	IDLE	-7.812	18.118
22562	15356	.645	337	404	-59.29	-3.85	0:10:13	47.680	421	IDLE	-7.971	17.728
23062	15859	.646	335	404	-58.59	-4.97	0:10:21	48.632	426	IDLE	-8.128	17.338
23562	16353	.647	333	403	-57.91	-4.91	0:10:30	49.596	430	IDLE	-8.288	16.944
24062	16866	.649	330	403	-57.25	-4.85	0:10:39	50.571	435	IDLE	-8.449	16.545
24562	17368	.650	327	403	-56.60	-4.79	0:10:47	51.556	439	IDLE	-8.613	16.141
25062	17871	.651	324	403	-55.96	-4.74	0:10:56	52.553	443	IDLE	-8.778	15.732
25562	18318	.654	323	405	-55.90	-4.21	0:11: 5	53.553	447	IDLE	-8.945	15.319
26062	18825	.655	321	404	-55.24	-4.71	0:11:14	54.566	452	IDLE	-9.114	14.899
26562	19331	.656	318	404	-54.59	-4.65	0:11:23	55.591	456	IDLE	-9.285	14.474
27062	19732	.660	316	406	-54.50	-4.12	0:11:33	56.620	460	IDLE	-9.457	14.043
27562	20292	.661	314	405	-53.82	-4.60	0:11:42	57.662	464	IDLE	-9.629	13.610
28062	20746	.664	312	407	-53.68	-4.08	0:11:51	58.710	468	IDLE	-9.798	13.173
28562	21263	.665	310	406	-52.98	-4.55	0:12: 1	59.771	472	IDLE	-9.969	12.718
29062	21773	.666	307	406	-52.32	-4.49	0:12:10	60.845	476	IDLE	-10.141	12.224
29562	22172	.671	306	409	-52.68	-3.51	0:12:20	61.916	479	IDLE	-10.308	11.726
30062	22636	.674	305	410	-52.46	-4.04	0:12:29	62.997	483	IDLE	-10.475	11.223
30562	23102	.677	303	411	-52.21	-4.04	0:12:39	64.084	487	IDLE	-10.643	10.711
31062	23514	.682	303	413	-52.45	-3.56	0:12:48	65.172	491	IDLE	-10.815	10.186
31562	23991	.684	302	414	-52.12	-4.09	0:12:58	66.270	494	IDLE	-10.990	9.650
32062	24411	.689	301	416	-52.28	-3.60	0:13: 8	67.369	497	IDLE	-11.168	9.100
32562	24490	.708	307	427	-36.79	-0.47	0:13:21	68.960	510	1.104	-11.380	8.505
33062	24945	.711	306	428	-36.85	-2.66	0:13:35	70.569	527	1.104	-11.641	7.849
33562	25232	.722	309	434	-37.79	-1.71	0:13:48	72.152	534	1.104	-11.870	7.212
34062	25669	.726	308	436	-37.93	-2.59	0:14: 1	73.741	546	1.104	-12.086	6.579
34562	26108	.730	307	437	-38.07	-5.20	0:14: 8	74.533	552	1.104	-12.300	5.988
34812	26322	.732	306	438	-38.16	-2.54	0:14:14	75.328	557	1.104	-12.406	5.705
35062	26539	.734	306	439	-38.25	-2.55	0:14:21	76.123	563	1.104	-12.512	5.418
35312	26753	.736	305	440	-38.34	-2.55	0:14:27	76.918	569	1.104	-12.617	5.126
35562	26969	.738	305	441	-38.43	-2.56	0:14:34	77.712	574	1.104	-12.722	4.829
35812	27175	.741	305	442	-38.56	-2.45	0:14:40	78.506	580	1.104	-12.827	4.522
36062	27363	.744	305	443	-37.78	-2.17	0:14:47	79.318	586	1.121	-12.943	4.190
36312	27550	.747	305	445	-36.21	-2.08	0:14:54	80.169	592	1.150	-13.074	3.821
36562	27729	.751	306	447	-34.97	-1.90	0:15: 1	81.054	599	1.176	-13.211	3.425
36812	27918	.754	306	448	-33.85	-1.94	0:15: 8	81.971	607	1.198	-13.353	2.966
37062	28127	.757	305	449	-33.92	-2.15	0:15:16	82.889	615	1.199	-13.480	2.426
37312	28307	.760	306	451	-18.64	-1.01	0:15:29	84.565	635	1.450	-14.043	0.984

TABLE 9.- SUMMARY OF SYNTHESIZED TRAJECTORY

Parameters	Initial cruise	Final cruise	Parameters	Initial cruise	Final cruise
Aircraft weight, lb	145,499	145,499	TAS, knots	481.00	481.00
Cost, $/n. mi.	2.233	2.233	IAS, knots	335.60	335.60
Energy height, ft	37,392	37,392	Groundspeed, knots	480.82	480.82
Altitude, ft	27,166	27,166	Mach number	.80606	.80606

	Fuel used, lb	Distance, n. mi.	Hr:min:sec	Cost, $	$/n. mi.
Climb	4500.83	115.99	0:17:57	431.42	3.72
Descend	644.75	85.46	0:15:38	170.65	2.00
Cruise	0	0	0: 0: 0	0	0
Total	5145.58	201.45	0:33:35	602.07	2.99

Landing weight = 144,854
Cruise and overall efficiency 0.000 25.543 lb/n. mi.
Cost, $\%_\lambda$ = 4.69
Number of iterations = 4

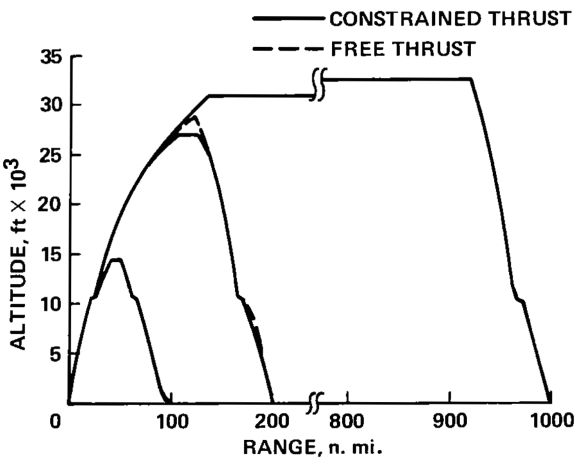
the cruise fuel efficiency and the overall fuel efficiency, both in pounds per nautical mile. It also shows the numerical value of $\%_\lambda$ and the number of iterations required to synthesize the trajectory.

RESULTS

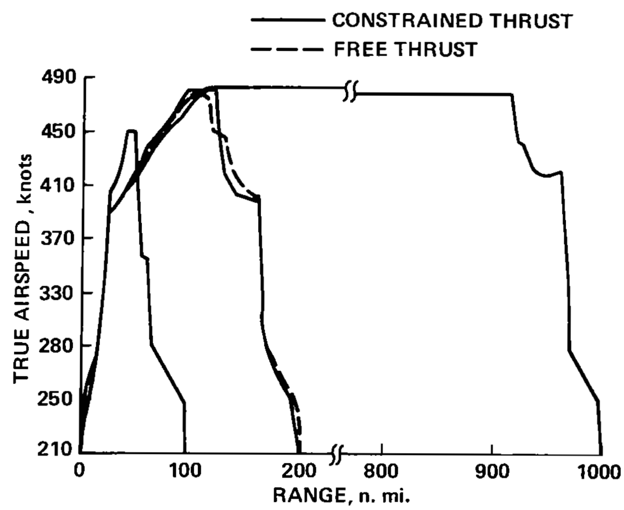
In this section, a selected class of trajectories is synthesized by the computer-implemented algorithm to determine the effect of selected parameters on: (1) the unit cost (in dollars per nautical mile), (2) the unit fuel consumption (in pounds per nautical mile), (3) the mission cost (in dollars), and (4) the total fuel consumption (in pounds). The parameters include (1) the range, (2) cost function (minimum cost versus minimum fuel), (3) mode of optimization (free thrust vs constrained thrust), (4) environmental conditions, such as wind and temperature variation (from standard atmosphere), (5) an ATC restriction placing a 250-KIAS limit on speed when flying at or below 10,000 ft, and (6) the takeoff weight. In all cases, the time cost is $500/hr and the fuel cost is $0.0626/lb.

The characteristic trajectories are shown in the following formats: (1) altitude and speed vs distance, and altitude vs speed; (2) unit operating cost and unit fuel cost vs range; (3) summary tables of the synthesized trajectories; and (4) tables showing the effect of speed limit, optimization mode, and takeoff weight on fuel consumption and mission cost.

Figure 9(a) shows four altitude profiles for minimal-cost trajectories as a function of range. The takeoff weight for these trajectories is 150,000 lb and the ranges are 100, 200, and 1,000 n. mi. The solid line is the constrained-thrust mode of optimization and the dashed line the free-thrust mode. In all four trajectories, the first part of the climb and the last part of the descent are nearly identical. The gradient of climb is steepest in the beginning, at an average of about 530 ft/n. mi. It is gradually reduced to about 260 ft/n. mi. at an altitude of about 10,000 ft, and eventually to 81 ft/n. mi. at the end of climb. The descent profiles start with an initial descent gradient of 251 ft/n. mi. This gradient is gradually increased to 370 ft/n. mi. at an altitude of about 10,000 ft and then gradually reduced to 210 ft/n. mi. at the end of descent.



(a) Altitude profile.



(b) True airspeed profile.

Figure 9.- Profiles for minimal-DOC trajectories.

At about 10,000 ft, both the climb and descent profiles are interrupted by short segments of almost level flight for all cases. This is a result of the 250 KIAS limit imposed on the trajectory below 10,000 ft. In climb, after reaching 10,000 ft, the speed boundary is removed, and most of the energy rate is used to temporarily accelerate the aircraft to the optimal speed at that altitude, thus resulting in practically level flight. After reaching the optimal speed, the climb is resumed. Similarly, in descent the aircraft enters the speed boundary at 10,000 ft and the negative energy rate is used for temporarily decelerating the aircraft in nearly level flight to 250 KIAS. After reaching that speed, the descent is resumed.

In the constrained-thrust trajectories, there is a short cruise segment in all cases. The cruise altitude for these segments increases with range until the optimal cruise altitude is reached. Beyond this range, the optimal cruise altitude is used regardless of range. For the short-range cases (100 and 200 n. mi.), the cruise altitude is practically constant. For the long-range case (1,000 n. mi.), the optimal cruise altitude increases at a

rate of 2.5 ft/n. mi. of cruise distance. It increases because the aircraft becomes lighter as a result of fuel burn during cruise.

The free-thrust trajectory for the 200-n. mi. range does not have cruise segment. The point at which the climb and descent segments join is the maximum energy point. It is also the highest altitude and the highest speed point. As shown in the figure, the 200-n. mi. range free-thrust trajectory differs from its corresponding constrained thrust trajectory only in the vicinity of the maximum energy point, specifically, at an altitude of 24,500 ft and above. Below this altitude, the optimal thrust, as determined by the algorithm, is the maximum thrust for climb and idle thrust for descent. Above this altitude, a gradual reduction in thrust occurs during climb until the maximum energy point is reached. At this point, the trajectory switches from climb to descent, i.e., the energy rate becomes negative. The initial part of the descent trajectory calls for a continuation in gradual thrust reduction until the 24,500 ft altitude is reached. At that altitude, the thrust is reduced to idle.

In the constrained-thrust mode, on the other hand, the thrust schedule calls for maximum thrust in climb, switching to cruise thrust at the end of climb, maintaining this thrust during cruise and switching to idle thrust at the end of cruise, and maintaining idle thrust during descent. For the 200 n. mi. range trajectories, a fuel saving is realized by optimizing over both thrust and speed as opposed to optimizing over speed alone.

Figure 9(b) shows the corresponding true airspeed profile. It calls for a high rate of acceleration during the first portion of the climb; this rate is gradually reduced for the remainder of the climb. The descent profile is symmetrical to the climb in the sense that it calls for a slow deceleration in the beginning of descent and gradually increases the deceleration until the final airspeed is reached. The cruise speeds for the short-range, constrained-thrust trajectories are practically constant. For the long-range flight, this speed is reduced slightly during cruise, in order to maintain the optimal cruise speed, which is a function of aircraft weight; the optimal cruise speed, as determined from the algorithm, decreases slightly as the aircraft burns fuel.

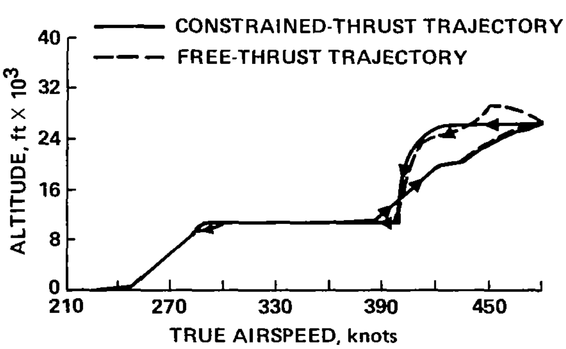
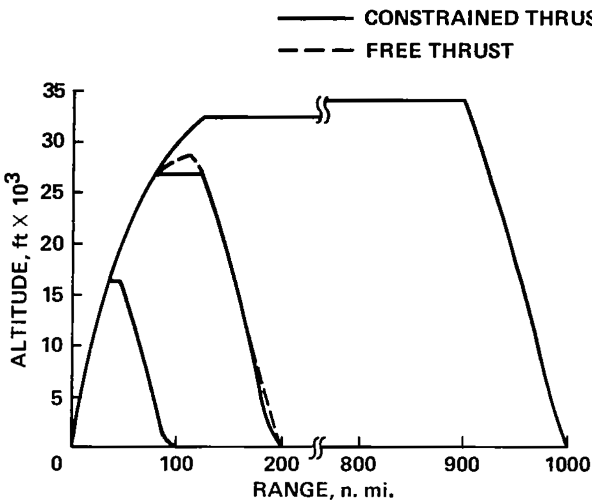


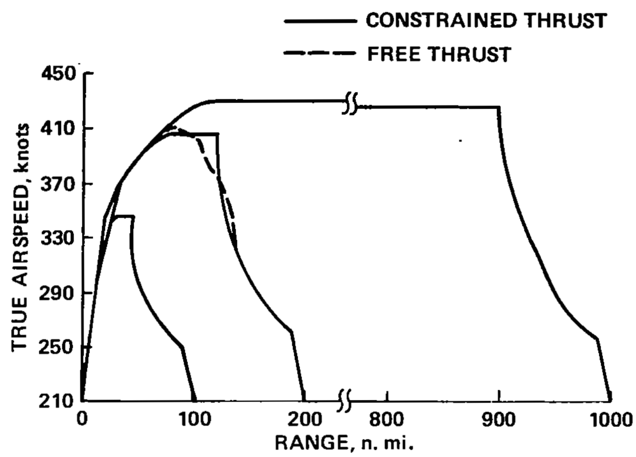
Figure 10.- Altitude-speed profile for 200-n. mi. minimal-cost trajectories.

Figure 10 shows the speed-altitude profiles for the 200-n. mi. constrained-thrust and free-thrust trajectories. In climb, most of the energy rate is used to accelerate the aircraft to the optimal speed at almost constant altitude. When the 250 KIAS limit is relaxed and in descent, most of the negative energy rate is used to decelerate the aircraft in nearly level flight as the 250 KIAS limit is reimposed.

Figures 11(a), 11(b), and 12 show the corresponding altitude profile,



(a) Altitude profile.



(b) True airspeed profile.

Figure 11.- Profiles for minimal-fuel trajectory.

speed profile, and speed altitude (VH) profile for the minimal-fuel trajectories for the same case as before. In comparison with the corresponding minimal-cost trajectories, the minimal-fuel trajectories call for a higher cruise altitude and lower cruise speed. The altitude profile has a steeper angle of climb and a shallower descent angle and the speed profile has a slower rate of acceleration and deceleration. In the VH profile, the acceleration and deceleration segments near an altitude of 10,000 ft are shorter or absent, as a result of lower optimal climb-out and descent speeds.

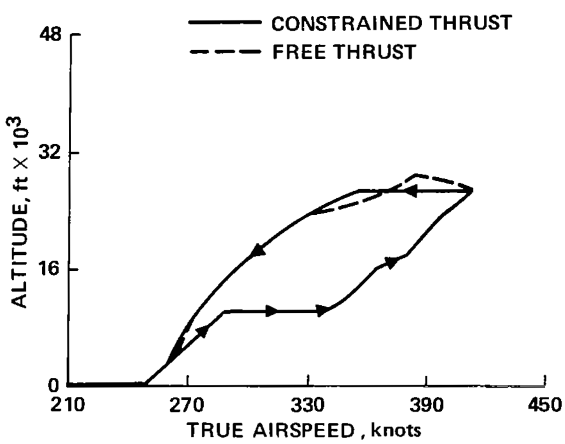
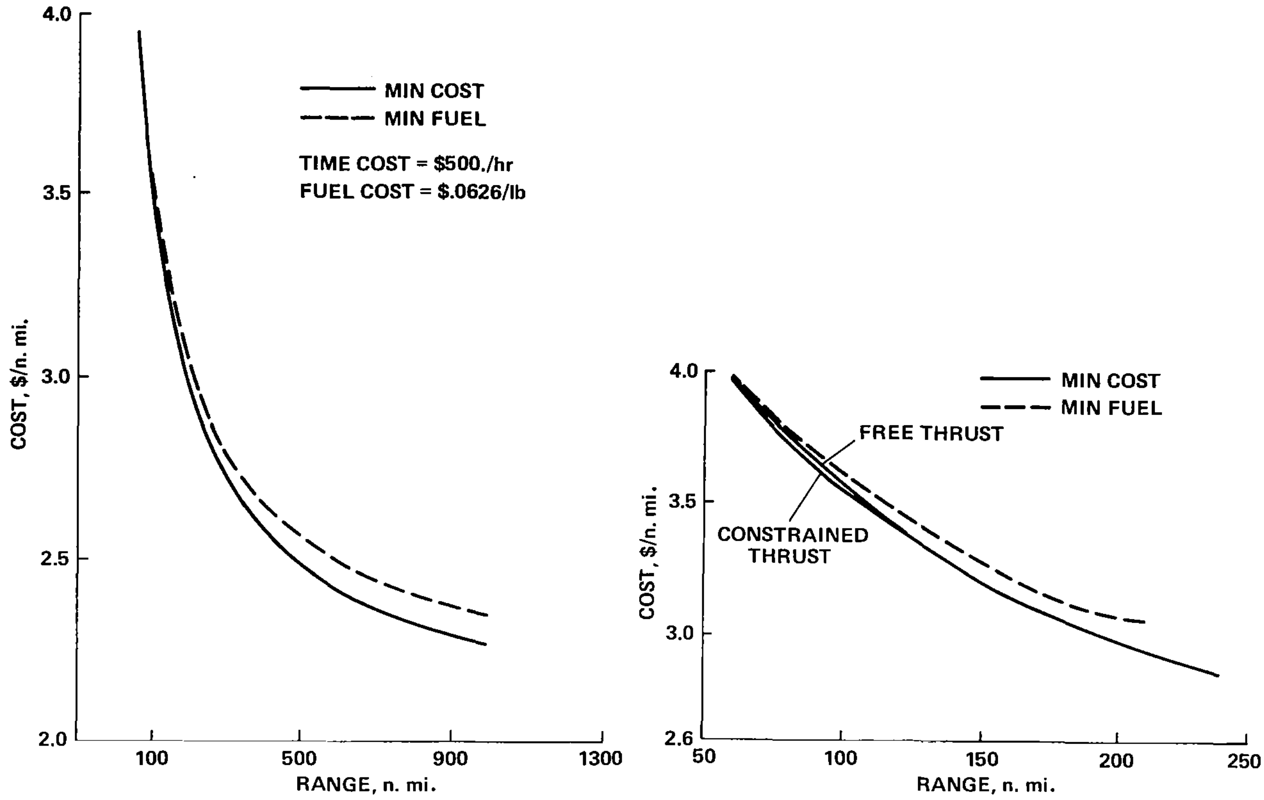


Figure 12.- Altitude-speed profile for minimal-fuel 200-n. mi. range trajectories.

Figure 13(a) shows the cost in dollars per nautical mile vs range for the minimal cost and the minimal-fuel trajectories. In the short-range flights (i.e., from 100 n. mi. to 250 n. mi.), the slope of the curves is very steep, implying that a slightly longer range flight will result in substantial saving in cost. For long-range flights (1,000 n. mi. or more), the slope of the curve is almost flat, implying that the cost is almost constant and practically independent of range. At longer ranges the vertical distance between the two curves becomes significant; for example, at 1,000 n. mi. it is \$0.08/n. mi. This distance represents the saving in cost realized by minimizing the cost instead of fuel consumption.

As previously discussed, the climb and descent profiles are restricted by the 250-KIAS limit at or below an altitude of 10,000 ft. In short-range



(a) Cost vs range.

(b) Cost for minimum-cost and minimum-fuel trajectories.

Figure 13.- Cost vs range: minimal-cost and minimal-fuel trajectories.

flights, this constraint is in effect for a large portion of the flight and therefore the difference in cost between minimal-cost and minimal-fuel trajectories is small. For very long range flights, on the other hand, most of the flight is spent in cruise (at optimal cruise altitude and speed) and, therefore, the difference between minimal-cost and minimal-fuel trajectories is the difference between their respective cruise costs.

Figure 13(b) is an expanded version of figure (13(a); it shows the difference in cost between the constrained-thrust mode and the free-thrust mode of optimization at short ranges. For the minimal-cost case, at a range of 100 n. mi., the cost saving between the two modes is $0.03/n. mi. At a range of about 135 n. mi., this difference diminishes to zero and the two curves merge, implying that for flights of 135 n. mi. or longer, no significant saving in unit cost is realizable by flying the free-thrust type of trajectory. In the minimal-fuel case, the fuel difference between the two modes is insignificant at all ranges.

Figure 14 shows the corresponding fuel efficiency versus range curves. The fuel saving in the minimum-fuel trajectory relative to the minimum-cost

trajectory is about 1 lb/n. mi. in long-range flight (more than 500 n. mi.) and 1.1 lb/n. mi. in short-range flights.

Table 10 summarizes the characteristics of selected minimal-cost trajectories for a takeoff weight of 150,000 lb. Column (1) gives the environment for the flight, namely the temperature deviation from standard temperature and the wind condition. Column (2) gives the mode of optimization (FT for free thrust and CT for constrained thrust). Column (3) shows the range and column (4) the time duration of flight. Columns (5) and (6) show the operating cost in \$/n. mi. and fuel consumption in lb/n. mi., respectively. The remaining columns give the cruise altitude, the climb distance, and the descent distance.

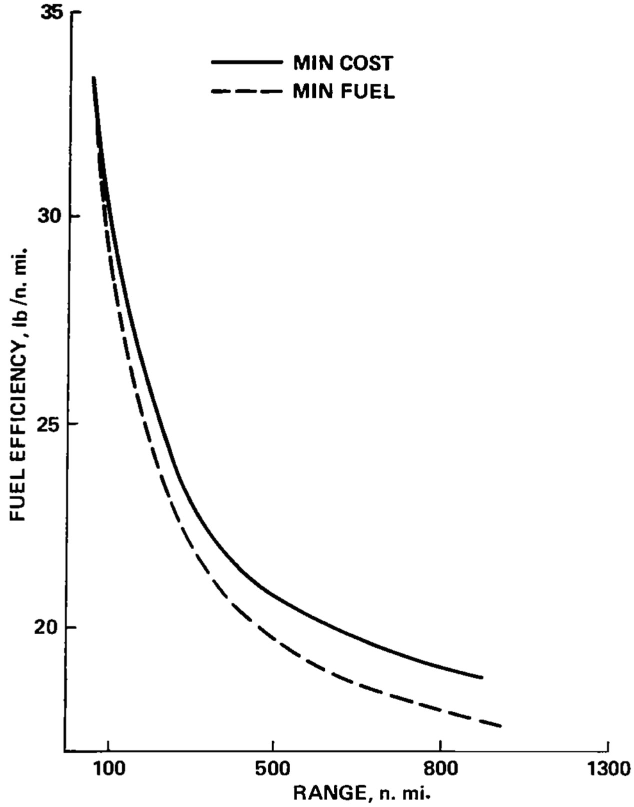


Figure 14.- Fuel efficiency vs range.

Table 10 shows that the differences in cost and fuel consumption between the free-thrust and the constrained-thrust modes of optimization for the 200-n. mi. range trajectories are \$0.04/n. mi. and 0.44 lb/n. mi., respectively, amounting to a 1% difference in cost and a 1.75% difference in fuel consumption. Reducing the temperature (in the free-thrust mode) at all altitudes results in insignificant cost and fuel reductions. However, raising it to 20° C at all altitudes increases the cost \$0.11/n. mi. and the fuel consumption 0.95 lb/n. mi; this is an increase of 2.3% in cost and of 3.75% in fuel consumption.

Next, consider the effect of wind. The wind profile used for the example trajectories of table 10 is shown in figure 15. Although it is the average profile in January over Cape Kennedy (ref. 5), the profile would be similar at other locations at the same latitude. It starts with a 10-knot wind at zero altitude that increases to 70 knots at an altitude of 36,000 ft; the increase is at a rate of 1-2/3 knots/1,000 ft. Above 36,000 ft, the wind decreases at a rate of 2-1/4 knots/1,000 ft. If the aircraft is flying against the wind, the unit cost increases \$0.27/n. mi., or 9.1%, and the fuel consumption increases 2.78 lb/n. mi., or 11%. If flying with the wind, the cost decreases \$0.16/n. mi., or 5.4%, and the fuel consumption decreases 1.82 lb/n. mi., or 7.2%.

Table 11 summarizes the characteristics of selected minimal-fuel trajectories corresponding to the minimal-cost trajectories shown in table 9. The takeoff weight for all the trajectories is again 150,000 lb. The differences in cost and fuel consumption between the free- and constrained-thrust modes

TABLE 10.- MINIMAL DIRECT OPERATING COST TRAJECTORIES: TAKEOFF WEIGHT 150,000 LB

Environmental condition		Thrust mode	Range, n. mi.	Time, hr:min:sec	Cost, $/n. mi.	Fuel, lb/n. mi.	Cruise altitude, ft	Climb distance, n. mi.	Descent distance, n. mi.
Standard temperature	No wind	CT	100	20:06	3.58	30.40	14,899	43.15	52.66
		CT	200	33:02	3.00	25.77	26,970	101.42	77.85
		FT	200	33:00	2.96	25.33	27,827	116.00	84.00
		CT	1000	2:13:07	2.28	18.78	30,819	135.76	85.38
+20° C		FT	200	34:07	3.07	26.28	25,007	122.35	77.65
-15° C		FT	200	32:54	2.96	25.31	31,467	112.50	89.51
Standard temperature	Headwind	FT	200	35:16	3.23	28.11	28,934	117.99	82.01
Standard temperature	Tailwind	FT	200	31:46	2.80	23.51	26,300	109.33	90.67

Note: CT = Constrained thrust, FT = Free thrust.

TABLE 11.- MINIMAL FUEL TRAJECTORIES: TAKEOFF WEIGHT 150,000 LB

Environmental condition		Thrust mode	Range, n. mi.	Time, hr:min:sec	Cost, $/n. mi.	Fuel, lb/n. mi.	Cruise altitude, ft	Climb distance, n. mi.	Descent distance, n. mi.
Standard temperature	No wind	CT	100	21:26	3.60	29.25	17,531	37.73	54.12
		CT	200	37:03	3.07	24.38	27,226	80.06	83.21
		FT	200	37:06	3.06	24.27	28,011	101.93	98.07
		CT	1000	2:29:14	2.36	17.76	33,185	121.07	103.51
+20° C		FT	200	37:00	3.11	25.06	27,273	112.38	87.62
-15° C		FT	200	36:00	3.04	24.05	32,330	95.31	104.69
Standard temperature	Headwind	FT	200	40:01	3.34	26.67	29,043	101.30	98.70
Standard temperature	Tailwind	FT	200	34:36	2.85	22.43	27,045	99.99	100.01

for the 200-n. mi. trajectories are \$0.01/n. mi. and 0.11 lb/n. mi., respectively; they are equivalent to \$1.08 and 23 lb of fuel. If the temperature is 15° C below the standard, cost and fuel reductions are \$0.02/n. mi. and 0.12 lb/n. mi., respectively. If the temperature is 20° C above the standard, the cost increases \$0.05/n. mi., or 1.6%, and the fuel consumption increases 0.79 lb/n. mi., or 3.3%. If the aircraft is flying against the wind, the cost increases \$0.28/n. mi., or 9%, and the fuel consumption increases 2.4 lb/n. mi., or 10%. If the aircraft is flying with the wind, the cost decreases \$0.21/n. mi., or 6.9%, and the fuel consumption decreases 1.8 lb/n. mi., or 7.6%.

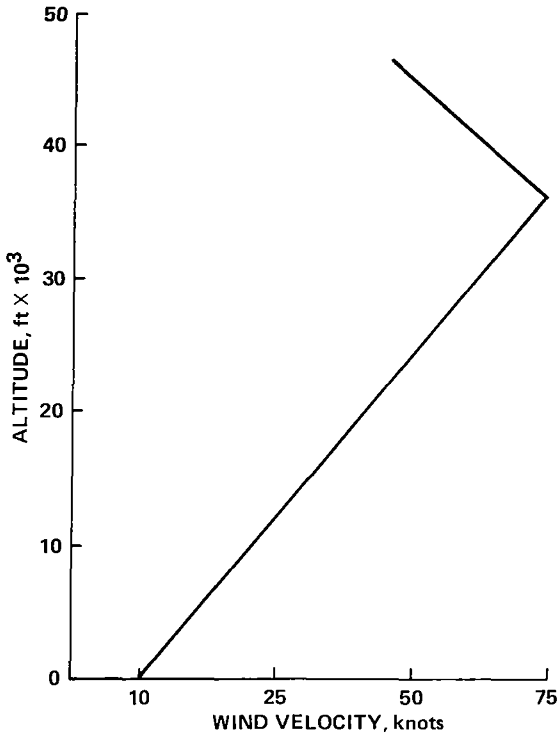


Figure 15.- Wind profile.

Finally, consider the effect of the 250-KIAS limit at or below an altitude of 10,000 ft on fuel use and mission cost (not shown in the tabulated results). The increase in cost due to this limit is essentially independent of range but strongly dependent on takeoff weight. At a takeoff weight of 180,000 lb, the speed constraint results in the consumption of an additional 133 lb of fuel; the corresponding increase in cost is \$18.02. At a takeoff weight of 150,000 lb, an additional 66 lb of fuel are consumed and the cost increase is \$9.77.

CONCLUSIONS

The approach presented here, which uses a structure of optimum trajectories suitable for airline operations, has yielded an efficient computer algorithm for calculating the trajectories. The algorithm can be incorporated into an airline flight planning system or can be used for determining the performance penalty of simplified onboard algorithm. The latter application is important at this time in view of the current effort by industry to develop onboard flight management systems.

The computer implementation of the algorithm can be used for investigating the trade-off between cost and fuel consumption and to give insight into the characteristics of minimal-fuel and minimum-cost trajectories. It was found that minimal-fuel trajectories, compared with minimal-cost trajectories, have steeper climb angles, higher cruise altitudes, lower cruise speeds, and shallower descent angles.

Optimizing the climb and descent with respect to airspeed only while constraining the thrust to a maximum in climb and to idle in descent, resulted in only a slight increase in total fuel consumption and operating cost compared with optimizing the climb and descent with respect to both airspeed and thrust. The actual differences in performance are strongly dependent on the propulsion and aerodynamic models. If the differences can be verified to be negligible, as in the example calculation, the optimization need only be done with respect to airspeed, thus reducing the computation time. That may be an important consideration in onboard implementation.

Although the emphasis in this report has been an off-line, open-loop computation, eventually the most important application of the algorithm will be its use in an onboard flight management system. This approach would allow the optimum trajectory to be updated with in-flight measurements of temperature and winds, resulting in closed-loop optimality throughout the flight.

Ames Research Center
National Aeronautics and Space Administration
Moffett Field, California 94035, January 2, 1980

APPENDIX

COMPUTER PROGRAM AND ITS USE

This appendix is addressed to users of this program. As a package, the program contains 1 main program, 39 subroutines (all of which are programmed in FORTRAN IV), and 3 sample input data sets. The program can be obtained from the centralized facility known as COSMIC, located at the Computer Software Management and Information Center, Barrow Hall, University of Georgia, Athens, Georgia, 30601.

The FORTRAN programs, a cross-reference between calling program and subroutines, a cross-reference of labeled commons, the input data sets, the FORTRAN units for input/output, and a dictionary of all FORTRAN programs are included.

Computer Programs

The computer programs may be conveniently divided into the main program and four groups of subroutines. The main program calls the appropriate subroutines to execute the main steps of the algorithm enumerated in figures 7 and 8. According to their functions, the four groups are (1) trajectory synthesis and optimization subroutines, (2) aerodynamics and propulsion subroutines, (3) environmental condition subroutines, and (4) general and supporting subroutines.

An alphabetical listing of all the subroutines, together with a one-phrase description of each, is shown in table 12. The most often used subroutines in each group will be considered.

In the first group, the subroutines MINF and MINF2 are used for minimizing a specified function by Fibonacci search over some given interval. The independent variable is searched within 1/144 of the given interval when using MINF and within 1/55, when using MINF2, where 144 and 55 are the eleventh and eighth Fibonacci numbers, respectively. These two, together with other appropriate subroutines, are used for generating the requisite boundaries and extremal points. The subroutine MINF, together with FBOUND, is used for generating (1) the speed boundaries designated as $V_{min,TD}$ and $V_{max,TD}$ in figures 4(b) and 4(c); and (2) with PILIMT for generating the lower EPR limit for climb optimization and the upper EPR limit for descent optimization. It is also used, together with FDRAG, for determining the minimum drag speed. Furthermore, it is used, together with FCLIMB and with FTHRST, for determining, respectively, the minimum cost with respect to V and that with respect to EPR setting for generating both the optimal climb and the optimal descent profile. It is used, together with FCOST, for generating the optimal cruise cost at a fixed altitude, i.e., for determining the value of λ by minimizing the cruise cost over V, as shown in equation (12). Finally, the subroutines

MINF, MINF2, and FOPT are used for generating the optimal cruise cost λ^* for all altitudes, as shown in equation (21b).

TABLE 12.- LIST OF PROGRAMS

Name	Brief description
MAINPROG	Overall algorithm
AT62	Compute atmospheric density, pressure, temperature, and speed of sound
CDRAG	Compute drag coefficient
CLIFT	Compute lift coefficient
CPMEPR	Read in engine data
CRUTBL	Read in cruise table
CRUZOP	Generate cruise table
ENGEPR	Compute thrust from engine data
EPR10	Auxiliary read/write program for CPMEPR
FBOUND	Compute drag or $T_{max} - D$
FCLIMB	Evaluate the cost function at constant EPR setting, for climb/descent
FCLMB2	Minimizes the cost function for climb/descent
FCOST	Evaluate cruise cost
FDRAG	Compute drag
FOPT	Compute optimal cruise speed
FTHRST	Evaluate cost function at constant speed
FULEST	Fuel estimation
ICLOCK	Change seconds into hours, minutes, seconds
JTRUNC	Truncate a unimodal series into a monotonically decreasing series
LSQPOL	Least-square polynomial fit
MATINV	Matrix inversion
MINF	Minimize a function over a given interval
MINF2	Minimize a function over a given interval
PCCOMP	Compute $\%_\lambda$ over $\lambda*$
PILIMT	Compute minimum EPR setting for climb and maximum EPR setting for descent
POLYE1	Evaluate a polynomial
SERCHD	Search an element in a monotonically decreasing series
SERCH1	Search an element in a monotonically increasing series
TRIM1	Compute thrust, etc., to keep the aircraft in trim
VALUE2	Extract a number from a two-dimensional array
UPDOWN	Generate the climb/descent profile
VOPTRJ	Compute fuel, time, and distance for the overall trajectory
WATEST	Weight estimation
WIND	Compute wind component along the wind axis
WINDIN	Read in wind model and compute wind shear for all altitudes
WLEFHV	Compute λ, etc., for given cruise weight
WRITE1	Auxiliary writing subroutine for VOPTRJ
XTRPL1	Extrapolate a point outside the lower end of a table
XTRPL2	Extrapolate a point outside the upper end of a table

Data Sets

INP.TRJ	Optimal trajectory specification
INP.WIND	Wind profile
INP.ENGINE	Engine data

In the same group, the subroutine CRUZOP is used for generating the cruise table (see table 3) and CRUTBL, for reading in the cruise table, if it is available. The subroutine FCLMB2 is used for generating the optimal cost for climb-descent trajectory optimization, at some fixed energy level. The subroutine UPDOWN is called to generate either the climb or descent profile, as specified in the calling program. The subroutine VOPTRJ is used to compute the fuel used, the distance traversed, and the time duration for the climb, cruise, descent, and the overall trajectory. The subroutine FULEST is called for estimating the climb or descent fuel, as specified in the calling program. The subroutine WATEST is used to estimate the landing weight. The subroutine WLEFHV is used to provide a mapping between λ, as a function of cruise weight and of $\%_\lambda$, and the corresponding cruise-energy height, fuel-flow rate, altitude, and ground speed. Finally, the subroutine PCCOMP is used to estimate the $\%_\lambda$ to achieve a given range R.

In the second group, the subroutine CLIFT is used for computing the lift coefficient as a function of Mach number, altitude, and angle of attack. The coefficients of polynomial-fitted curves for determining the lift coefficient are stored in the arrays CL216, CL217, CL218, CL219, CL17, and CL18 in the subroutine. The subroutine CDRAG is used for computing the drag coefficient as a function of Mach number and lift coefficient. The coefficients of polynomial fitted curves for determining the drag coefficient are stored in the arrays CD223, CD224, CD10, and CD11 in the subroutine. The subroutine ENGEPR is used for computing the thrust and fuel flow as functions of altitude, Mach number, temperature variation from standard atmospheric condition, and EPR setting. The thrust is computed from tabulated engine data. The fuel-flow rate is computed from one of two alternatives, depending on the user's choice. The first is obtained from evaluating polynomials, the coefficients of which were obtained by fitting a set of curves supplied by the engine manufacturer, Pratt and Whitney. These coefficients are stored in the array WF in the ENGEPR subroutine. The fuel-flow rate thus obtained is a function of EPR setting, altitude, and Mach number, corrected for temperature and pressure. The second alternative, which requires considerably more data storage than the first, is to compute the fuel-flow rate from tabulated TSFC (thrust specific fuel consumption) data. The fuel-flow rate is a product of thrust and TSFC, corrected for temperature. The tabulated TSFC data are functions of thrust, Mach number, and altitude. The fuel-flow rate thus obtained takes into consideration average bleed losses. Therefore, this method generally gives fuel-flow rates that are 2% - 4% higher than those determined by the first method. This subroutine has three satellite subroutines (functions), namely, VALUE2, XTRRPL1, and XTRPL2. The first is used for interpolating-extrapolating a point in a two-dimensional array, the second and third for extrapolating, respectively, a point beyond the smallest element and the largest element in the array. The subroutine CPMEPR is used for reading in the tabulated engine data, from the data set labeled INP.ENGINE. The subroutine TRIM1 is used for determining the thrust for maintaining constant speed and level flight.

In this package, the data set INP.ENGINE and the coefficients for determining the fuel-flow rate stored in the subroutine ENGEPR are for the Pratt-Whitney JT8D-7 engines. Also, the aerodynamic model, stored permanently as data in the CLIFT and CDRAG subroutines, is for the Boeing 727-100 aircraft. If other aircraft were used, these coefficients and the data set INP.ENGINE

would have to be replaced to reflect the aircraft and engines under consideration.

In the third group, the subroutine AT62 is used for generating the atmospheric density, pressure, temperature, and speed of sound each as a function of altitude. The values of these four quantities for some fixed altitude is given in a Boeing handbook (ref. 7). The subroutine WINDIN is used for reading the wind profile, shown in figure 15, from the data set INP.WIND. It is also used for computing the wind shear from the profile just read in. The subroutine WIND is used for computing the component of wind and also the component of wind shear projected along the principal axis of the aircraft.

If any other wind model were used, the subroutine WINDIN would have to be replaced to reflect the change.

In the fourth group, the subroutines include SERCH1, SERCHD, JTRUNC, LSQPOL, POLYE1, MATINV, ICLOCK, and WRITE1. A brief explanation of their use is shown in table 12; a detailed explanation is given in the dictionary at the end of this appendix.

CROSS-REFERENCES

Table 13 provides a cross-reference between calling programs and the subroutines being called. Each row lists the subroutines called by the appropriate calling subroutine. For example, FOPT calls the subroutines AT62, CDRAG, ENGEPR, FBOUND, FCOST, MINF, and WIND. Each column lists all the calling programs for the appropriate subroutine. For example, ENGEPR is called by the CRUZOP, FBOUND, FCLIMB, FDRAG, FOPT, FTHRST, PILIMT, and TRIM1 subroutines.

Table 14 provides a cross-reference between the labeled commons and subroutines in which these commons are used. Each row lists all the labeled commons used by the appropriate subroutine, and each column lists all the subroutines in which the appropriate labeled common is used. Within the appropriate subroutine, these commons are declared in alphabetical order.

INPUTS

Inputs to this program consist of (1) specification of desired trajectory, (2) wind profile, and (3) engine data.

Inputs for trajectory specification are shown in table 15. The input consists of five cards if the cruise table is not available and three cards if it is available. Card 1 contains logical values specifying whether the cruise table is available; the mode of optimization, either constrained thrust or free thrust; whether the wind profile is to be used; whether the fuel-flow rate or alternative should be obtained from the Pratt-Whitney curve-fitted data or from data source; whether the trajectory is restricted to a maximum speed of 250 KIAS at or below an altitude of 10,000 ft, or is unrestricted;

TABLE 13.- CROSS REFERENCE OF SUBROUTINES AND CALLING PROGRAMS

SUBROUTINES CALLED

CALLING PROGRAM	AT62	CDRAG	CLIFT	CPMEPR	CRUTBL	CRUZOP	ENGEPR	EPRIO	FBOUND	FCLIMB	FCLMB2	FCOST	FDRAG	FOPT	FTHRST	FULEST	ICLOCK	JTRUNC	LSQPOL	MATINV
MAINPROG				X	X	X										X				
CDRAG																				
CLIFT																				
CPMEP1								X												
CRUTBL	X															X	X			
CRUZOP	X						X		X			X		X				X	X	
ENGEPR	X																			
FBOUND		X					X													
FCLIMB	X	X					X													
FCLMB2										X			X		X					
FCOST																				
FDRAG	X	X					X													
FOPT	X	X					X		X			X								
FTHRST							X													
FULEST																				
LSQPOL																				X
PCCOMP																				
PILIMT							X													
TRIM1	X	X	X				X													
UPDOWN	X										X						X			
VALUE2																				
VOPTRJ	X																			
WATEST																				
WIND																				
WINDIN																				
WINDMOD																				
WLEFHV																		X		
WRITE1																	X			

CALLING PROGRAM	MINF	MINF2	PAGE	PCCOMP	PILIMT	POLYE1	SERCHD	SERCH1	TRIM1	UPDOWN	VALUE2	VOPTRJ	WATEST	WIND	WINDIN	WINMOD	WLEFHV	WRTEL1	XTRPL1	XTRPL2
MAINPROG			X	X						X		X	X							
CDRAG						X		X												
CLIFT						X														
CPMEP1																				
CRUTBL			X				X	X						X	X					
CRUZOP	X	X	X					X						X	X					
ENGEPR						X		X			X									
FBOUND																				
FCLIMB											X			X						
FCLMB2	X				X									X						
FCOST									X											
FDRAG																				
FOPT	X													X						
FTHRST																				
FULEST																	X			
LSQPOL						X														
PCCOMP								X												
PILIMT																				
TRIM1																X				
UPDOWN			X					X						X						
VALUE2								X											X	X
VOPTRJ			X					X									X	X		
WATEST						X		X									X			
WIND								X												
WINDIN			X																	
WINDMOD								X												
WLEFHV							X							X						
WRITE1																				

TABLE 14.- CROSS REFERENCE OF LABELLED COMMONS AND SUBROUTINES

USING THESE COMMONS

LABELLED COMMON

CALLING PROGRAM	B1	BOEING	BWIND	CLIMB	COST	CRURNG	CRUZ	DRAGMN	ENERGY	ENGIN	EPSILN	IFLAP	III	IO	OPT	RITE	TRIM2	TTRIM1	VTRCRU	VUPDWN	WINDTD	WINDY
MAINPROG				X	X	X		X	X		X		X	X	X				X			X
CDRAG												X										
CLIFT												X					X					
CPMEPR		X																				
CRUTBL					X									X					X			X
CRUZOP				X	X			X	X				X	X	X				X			X
ENGEPR		X			X																	
FBOUND					X			X					X									
FCLIMB		X		X	X		X	X	X						X							X
FCLMB2				X	X	X	X	X	X		X		X	X	X							X
FCOST					X			X														X
FDRAG				X	X			X	X				X									
FOPT					X			X					X		X							X
FTHRST				X	X		X	X	X						X							X
FULEST																			X			
PILIMT				X				X														
TRIM1	X				X			X		X		X				X	X	X				
UPDOWN				X	X		X	X	X					X	X					X		X
VOPTRJ					X	X			X										X	X		X
WATEST						X													X	X		
WIND																					X	
WINDIN																					X	
WINMOD	X		X																			
WLEFHV						X			X					X					X			X

TABLE 15.- INPUTS FOR SPECIFYING THE DESIRED TRAJECTORY

Card	Symbol	Explanation	Format	Calling program
1	ITAB	Availability of cruise table	20I4	Mainprog
		= 0 cruise table is not available and must be generated		
		≠ 0 cruise table is available and will be read in		
	IPRINT	Trial-descent profile output		
		= 0 print trial-descent profile		
		≠ 0 do not print trail-descent profile		
	IOPARM	Optimization mode		
		= 0 optimizing over speed only or constrained thrust mode		
		≠ 0 optimizing over both speed and thrust, or free thrust mode		
	IWIND	Effect of wind		
		= 0 generate the cruise table and/or synthesize the trajectory without taking wind into consideration		
		≠ 0 use the supplied wind profile to generate the cruise table and/or synthesize the trajectory		
	MFGR	Source of fuel-flow-rate data		
		1 use the Pratt-Whitney curve fitted data to determine fuel-flow rate		
		2 use the typical tabulated data to determine the fuel-flow rate		
	IBTALB	= 0 do not print the tabulated engine data		
		≠ 0 print the tabulated engine data		
	ISPLMT	= 0 no ATC restriction on speed		
		≠ 0 speed is limited to 250 KIAS at or below 10,000 ft altitude		

TABLE 15.- CONCLUDED

Card	Symbol	Explanation	Format	Calling program
2	FC	Fuel cost in $/lb	8F10.2	CRUZOP
	TC	Time cost in $/hour		
	DTEMPK	Temperature variation from standard atmospheric temperature, deg K (or C)		
	PSIA	Aircraft heading in degrees, heading north as 0°		
		this input is ignored in the absence of wind		
3	W	Heaviest cruise weight to be used for generating the cruise table, lb	8F10.2	CRUZOP
	WN	Lightest cruise weight to be used for generating the cruise table, lb		
	DEW	Decremental weight, lb, i.e., W will be used to generate the first page of the table, then W - DEW for the second page, etc., until WN is reached		
4	W	Aircraft takeoff weight, lb	8F10.2	CRUTBL
	RANGE	Desired range of the trajectory to be synthesized, nm		
5	HTO	Initial altitude (or takeoff altitude), ft	8F10.2	CRUTBL
	VTO	Initial speed (or takeoff speed), KIAS		
	HOLNDG	Final altitude (or landing altitude), ft		
	VOLNDG	Final speed (or landing speed), KIAS		

and other printing (computer output) options. Card 2 contains the numerical value for fuel cost in dollars per pound, time cost in dollars per hour, temperature variation from standard atmospheric condition in degrees Celsius, and aircraft heading in degrees from north. Card 3 specifies the heaviest cruise weight, the lightest cruise weight, and incremental weight to be used for generating the cruise table. Card 4 contains the value for the takeoff weight in pounds and the range of the trajectory in nautical miles. Finally, card 5 contains the value for the takeoff altitude in feet, takeoff speed in KIAS, landing altitude in feet, and landing speed in KIAS. Cards 2 and 3 contain input for generating the cruise table. Therefore, if the cruise table is not available, all five cards are required for transmitting the required quantities for synthesizing the trajectory. If the cruise table is available, only cards 1, 4, and 5 are required.

These quantities are transmitted to the program by the FORTRAN statement. READ(5, fmt) S1, S2 . . . where fmt is the appropriate format listed in column (4) (in the table) and S1, S2 . . . are the symbols listed in column (2). The program containing this READ(5, fmt) . . . statement is listed in column (5).

A typical data set containing all five cards is included in the package under the label of INP.TRJ.

Wind Input

The wind model is shown in figure 15. Input data for this profile consists of 24 sets of numbers, each set of which represents one point in the curve. Within each set there are two numbers: the first number is the value of the wind heading ψ_W and the second is the wind velocity V_W. The whole profile is packed into two cards, with 13 sets of numbers in the first and 11 in the second, as shown in table 16. The altitude is implied by the position of the set. The first set of numbers is for 0 altitude, the second set for 2,000 ft, and the last set for 48,000 ft, incremented in at 2,000-ft intervals.

TABLE 16.- INPUT CARDS FOR WIND PROFILE DATA SET INP.WIND

```
DATASET.FSTHQL.WINDINP1,,,,CARD,,REPLACE

100  133  167  200  233  267  300  333  367  400  433  467  500
533  567  600  633  667  700  654  609  563  518  473

%ENDDS
```

The input format is 13(F2.0, 1x, f3.1), where ψ_W is in tens of degrees and V_W in knots. For example, the second set of numbers in the first card is ---133, which stands for 0° heading and 13.3 knots at an altitude of 2,000 ft, (where – is for blank space). That is, the first two blank spaces are for zero by default and ψ_W = 00.X 10 = 0°, and since it is the second set, the altitude is 2,000 ft. A 0° wind heading stands for wind coming from the north.

This set of data is transmitted to the program by the FORTRAN statement READ(7, fmt) (PSIW(I), VW(I), I=1, 24). The subroutine containing this statement is WINDIN.

Engine Data and Mach-Corrected Drag Data Input

The tabulated data in the data set INP.ENGINE include (1) idle thrust (in pounds) as a function of altitude (in feet) and Mach number; (2) its corresponding fuel-flow rate in pounds per hour; (3) maximum continuous EPR as a function of both total air temperature, TAT (°C), and altitude; (4) corrected thrust (thrust:pressure ratio) as a function of EPR setting and Mach number; (5) effect of Mach number on drag, as a function of C_L; and (6) fuel-flow rate as a function of thrust, Mach number, and altitude.

These data are transmitted to the program by the FORTRAN statement READ(9, fmt).... the subroutine containing this statement is EPRIO and the subroutine that calls EPRIO repeatedly to read in all the engine data is CPMEPR.

FORTRAN Unit for Input/Output

The input/output unit for this program is shown in table 17. The first column is the unit for transmitting the appropriate quantity, i.e., READ(N, fmt) if it is an input and WRITE(N, fmt) if it is an output, where N is the appropriate unit number. The second and third columns specify whether it is an input or an output, or both, and the last column specifies the quantities being transmitted. Units 5, 7, and 9 have already been explained as inputs. Unit 6 is for writing outputs shown in tables 3 through 9 and other output quantities. The subroutine containing the WRITE(6, fmt)... statement for generating table 3 is CRUZOP; table 4, CRUZOP or CRUTBL; tables 5 and 6, CRUTBL; tables 7 and 8, UPDOWN; and for generating table 9, VOPTRJ. Unit 8 can be used for either writing the cruise table if it is being generated or for reading in the cruise table if it is available. The subroutines containing the WRITE(8) and READ(8) statements are, respectively, CRUZOP and CRUTBL. The input and output for this unit are unformatted.

TABLE 17.- INPUT/OUTPUT UNIT FOR COMPUTER PROGRAM

Unit	Input	Output	Comment
5	x		Trajectory specification
6		x	Computer outputs, see tables 3 through 9
7	x		Wind input
8	x	x	Cruise table
9	x		Engine data

DICTIONARY ENTRY OF SUBROUTINES

The remainder of this appendix contains dictionary entry for all FORTRAN programs listed in table 12. Each entry usually consists of (1) a functional description of the subroutine, (2) equations used in deriving certain quantities as appropriate, (3) input and output, as appropriate, and (4) an explanation of symbols frequently used in the subroutine.

Symbols containing a string of capital letters are always FORTRAN symbols. A single capital letter without any subscript is a FORTRAN symbol unless otherwise defined in the text, in the symbol table, or in the subroutine under discussion.

MAINPROG

This program calls the appropriate subroutines for synthesizing a fixed-range, optimal trajectory, as outlined in the flowcharts shown in figures 7 and 8. The major steps are:

1. Call CPMEPR for reading in the tabulated engine data.

2. Read in card 1, in table 15.

3. If ITAB = 0, the cruise table is not available, therefore call CRUZOP for generating the cruise table; if ITAB $\neq$ 0, the cruise table is available, so proceed.

4. Call CRUTBL for reading in the cruise table if it is available; skip this part if it is being generated in step (3) and then read in card 4, which contains the value of takeoff weight and range. Use these values to estimate the cruise weight W_c and compute its corresponding optimal cruise cost $\lambda*$, optimal cruise energy, E*, etc.

5. Compute the maximum value of $\%_\lambda$ for generating the shortest range trajectory. It is the least of 50%, or the values of $\%_\lambda$ corresponding to 10,000-ft cruise altitude for the two cruise weights closest to W_c, as contained in the cruise table.

6. Compute the cruise weight WCRUZ, which is specified by $\%_\lambda$ (determined in the last step). It is computed by subtracting the climb fuel from the takeoff weight. This climb fuel is estimated by calling the subroutine FULEST.

7. Call UPDOWN with ICLIMB set equal to 1 to generate the climb profile.

8. Call WATEST (weight estimate) to estimate the landing weight and the cruise weight at the beginning of descent.

9. Call UPDOWN with ICLIMB set equal to 2 to generate the trial descent trajectory.

10. Call VOPTRJ to generate the trial overall trajectory. At this stage, the descent and the landing weight are completely known.

11. Call UPDOWN with ICLIMB still set equal to 2 to generate the final descent profile.

12. Call VOPTRJ to generate the final overall trajectory.

13. Call PCCOMP to test the range of the synthesized trajectory against the desired range. If it is not within the preselected distance error, compute the new $\%_\lambda$ and go to step (6). If so, the required trajectory has been synthesized.

Symbols

ICLIMB if = 1, generate climb profile
if = 2, generate descent profile

ITAB, IPRINT, IOPARM, IWIND, MFGR, IBTABL, ISPLMT (see table 15)

ISPLIZ if 0, trajectory contains a cruise segment at optimal cruise altitude. This is the flag for adding in a cruise segment for the climb-descent type of trajectory.
if not 0, trajectory is not of the climb-cruise-descent type

PC $\%_\lambda$

RANGE desired range

TDIST ground-track distance of synthesized trajectory

WCRUZ in statement 81, cruise weight as a function of range; in statements 190 and 191 + (1), initial cruise weight or cruise weight at the end of climb; in statements 350 - (1) and 211, final cruise weight, or the weight at the beginning of descent

WCRUZF final cruise weight

WLNDG landing weight

AT62

This subroutine generates the atmospheric density (in lb sec^2/ft^4), atmospheric pressure (in lb/ft^2), atmospheric temperature (in °K), and speed of sound (in ft/sec) at a given altitude H (in ft). These four outputs are stored in ANS(1), ANS(2), ANS(3), and ANS(4), respectively.

CDRAG (MACH, CL, GEAR, DF, CD)

This subroutine computes the drag coefficient CD given a Mach number MACH, lift coefficient CL, flap angle DF, and landing gear position (up or down). The drag coefficient consists of two terms, CD(basic) and CD(gear), both of which are functions of CL, DF, and MACH. The values of both are stored as coefficients of polynomials, which are obtained by curve-fitting the appropriate data from the Boeing Handbook (ref. 7). The coefficients for CD(basic) are stored in the arrays CD223 and CD224, and for CD(gear) in the arrays CD10 and CD11.

Symbols

GEAR if 0, gears up
if not 0, gears down

DF flap angle, deg

CLIFT

This subroutine computes the lift coefficient CL for given Mach number MACH, altitude H, angle of attack ALPHAP, flap deflection DF, and gear position GEAR. The lift coefficient consists of four terms.

$$CL = CL\ (\text{basic}) + CL_o + CL_\alpha \alpha + CL_{gear}$$

The first term is the basic lift coefficient; it is expressed as a polynomial in α for different flap angles, namely, 0°, 2°, 5°, 15°, 25°, 30° and 40°. Each of these polynomials was obtained by fitting the appropriate curve, one per a specific value of flap angle. The coefficients of these polynomials are stored in the array CL216. The value of this term is checked against the buffet boundary function, a polynomial in Mach number, the coefficients of which are stored in array CL217.

The second term is the deviation from the basic airplane lift coefficient at $\alpha = 0$ due to Mach number and altitude. It is expressed as the difference between CL_o at the appropriate Mach number and at zero Mach number. The former is expressed as a polynomial in Mach number with altitude as parameter, namely at altitudes of 0, 20,000, and 40,000 ft. The coefficients of the polynomials are stored in the array CL218. CL_o at MACH = 0 has a value of 0.0175 and this value is subtracted from CL_o at the Mach number of interest to obtain the required CL_o correction term.

The third term corrects the basic lift coefficient for variations (due to Mach number and altitude) in the lift curve slope at $\alpha = 0$. It is expressed as the difference between the partial derivative of CL with respect to α at the appropriate Mach number and at zero Mach number. The former is expressed as a polynomial in Mach number with altitude as parameter. Its coefficients are stored in the array CL219. At MACH = 0, CL_α has a value of

0.088, which is subtracted from CL_{α} evaluated at the current value of MACH to obtain the required CL_{α} correction term.

The fourth term is the increment of lift coefficient due to deployment of the main and nose gears, expressed as a function of α, flap setting, and Mach number. If the gears are up, it is zero. If the flaps are down, its value is computed from the polynomial in α with flap-angle deflection as a parameter. The coefficients of the polynomial are stored in the array CL17. If the flaps are up, then it is a function of Mach number. It has a value of 0.02 if the Mach number is less than or equal to 0.4. If the Mach number is greater than 0.4, then CL_{gear} is computed from a polynomial in Mach number. The coefficients of the polynomial are stored in the array CL18.

CPMEPR

This subroutine reads in the tabulated engine data from the data set INP.ENGINE. The data include:

1. Idle thrust and idle fuel flow each as a function of both altitude and Mach number, Table 18Q001.

2. Maximum continuous EPR as a function of temperature and altitude, Table 24J011.

3. Thrust vs EPR as a function of Mach number, Table 184001.

4. Mach number effect on drag, that is CD as a function of CL and Mach number, Table 24B003.

5. Weighted average TSFC (thrust specific fuel consumption) as a function of altitude, thrust, and Mach number, Table 27273A.

CRUTBL

This subroutine (1) reads in the cruise table if it is available; (2) prints out the table of $d\lambda/dE$ coefficients (table 6) after both tables are read in from unit (8); (3) reads in the takeoff weight W and the desired range R from unit (5); (4) prints out the table of optimal cruise energy, speed and altitude vs weight (table 4), and generates and prints out the table of cruise quantities as a function of cruise distance (table 5).

<u>Symbols</u>

EOPT, EOPTS(.)	optimal cruise energy
EPRTAR, PISTRS(.)	the corresponding EPR setting
FUELDT, FUELFL(.)	the corresponding fuel-flow rate
HSTAR, HSTARS(.)	the corresponding optimal cruise altitude

LAMBDA, LAMBS(.)	the corresponding minimal cruise cost
MSTAR, MSTARS(.)	the corresponding Mach number

CRUZOP

This subroutine reads in card 2 and card 3 (see table 15) and then generates the cruise table (see table 3).

The table consists of pages, each for a specific cruise weight. Each page includes the optimal cruise altitude, speed, minimal cruise cost, the corresponding EPR setting, and the optimum cruise energy. There are two sets of these quantities for the specified cruise weight. The first set is obtained from minimizing equation (21) over both altitude and speed and the second from minimizing the fuel-flow rate $\dot{f}$ over both altitude and speed.

Each row in the table is for a specified altitude. For each altitude, two sets of these quantities were also obtained, one from minimizing equation (12) over speed and the second from minimizing $\dot{f}$ over speed.

A page of the cruise table is generated with the following steps:

1. For some given altitude h:

   a. Determine the minimum drag speed V_d (see fig. 4).

   b. Compute the maximum thrust T_{max} available for this speed and altitude.

   c. If T_{max} is greater than D at V_d proceed; otherwise, the ceiling has already been reached and proceed to (2).

   d. Compute the lower and upper permissible speeds $V_{min,TD}$ and $V_{max,TD}$, where the maximum thrust curve intersects the drag curve.

   e. Minimize equation (12) over V.

   f. Minimize $\dot{f}$ over V.

2. Minimize equation (21b) over both h and V.

3. Minimize $\dot{f}$ over both h and V.

<u>Symbols</u>

DEW, DTEMPK, FC, PSIA, TC, W, WN	see cards 2 and 3, table 15
A	speed of sound, ft/sec

E, DE	energy height, in ft, and incremental energy
FA, FB	lower and upper speed limits $V_{min,TD}$ and $V_{max,TD}$ (see fig. 4)
H, DEH	altitude, and incremental altitude, ft
P	atmospheric pressure, lb/ft^2
RATIO	pressure ratio P/P_O, where P_O is the atmospheric pressure at sea level and P is the static pressure at a given altitude
TEMPK	atmospheric temperature, K
ICOST	if = 1, the cruise cost is obtained by minimizing equation (12) if = 2, the cruise cost is obtained by minimizing $\dot{f}$
FDTOPT(1), MOPTAS(1), OPTALT(1)	minimal cost, its corresponding speed and altitude obtained by minimizing equation (12) over both V and H
MNCOST(1)	minimal cost obtained by minimizing over equation (12) for some fixed altitude
FDTOPT(2), MOPTAS(2), OPTALT(2)	minimal cost, its corresponding speed and altitude obtained by minimizing $\dot{f}$ over both V and h
MNCOST(2)	minimal cost obtained by minimizing $\dot{f}$ for some fixed altitude
ANS(.)	see AT62
EOPTS(.), EPRSTR(.), FUELFL(.), HSTARS(.), LAMBS(.), MSTRS(.)	see symbol in CRUTBL
EECRUZ(.,.), FFCRUZ(.,.), LLCRUZ(.,.)	the energy height, fuel-flow rate, and cruise cost corresponding to HHCRUZ(.,.)
HHCRUZ(.,.)	altitude as row index for given cruise weight
WS(.), WSS(.)	cruise weight, lb

ENGEPR

This subroutine generates the thrust THRST and the fuel-flow rate FDOT for some given altitude H, Mach number MAKNO, and EPR setting EPRX. The fuel-flow rate is obtained from evaluating polynomials if MFGR is set equal to 1 and from tabulated TSFC data if it is set equal to 2.

The maximum continuous EPR is primarily a function of total air temperature T_a and a weak function of altitude. Figure 16 shows this relationship for the JT8D-7A engine. A detailed discussion of these engine characteristics can be found in the operations instruction (ref. 8).

First, EPRMX, the maximum continuous EPR is obtained from the table. Total air temperature is obtained from the relation,

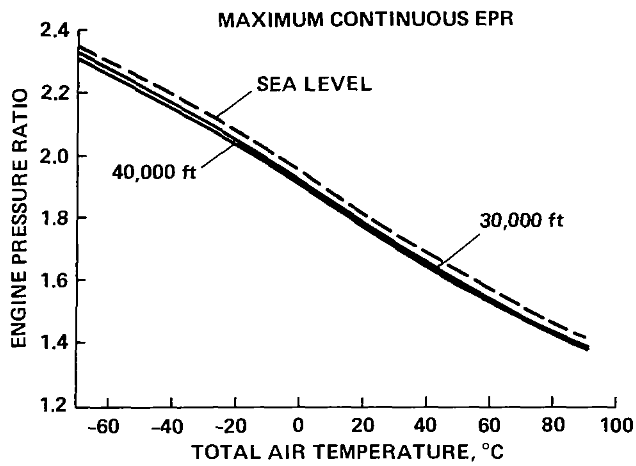


Figure 16.- Maximum continuous EPR.

$$T_a = T\left[1 + \frac{\gamma - 1}{2}(\text{MAKNO})^2\right] - 273.15 \quad (°\text{C})$$

where T is the ambient temperature and $\gamma = 1.4$ is the ratio of specific heats. Next, the EPR setting is limited to $\text{EPRX} \leq \text{EPRMX} - 0.1$ for cruise, $\leq$ EPRMX for climb or descent. Second, the value of F_n/δ, which is a function of EPR setting EPRX and Mach number MAKNO, is obtained from the table. With known F_n/δ, value of thrust, THRST, is obtained from the relation,

$$\text{THRST} = 3(\delta)(F_n/\delta)$$

where the pressure ratio δ is defined as

$$\delta = \frac{P}{P_o}$$

and where P is the atmospheric pressure corresponding to altitude H, and P_o is the atmospheric pressure at sea level. A factor of 3 is used since there are three engines.

Finally, if MFGR is set equal to 1, the fuel-flow rate is computed from the relations,

$$\text{FDOT} = 3 * \text{WFC} * \delta * \text{KC}$$

$$\text{KC} = 0.00223181\ T_a + 0.9675897$$

$$\delta_a = \delta\left(1 + \frac{\gamma - 1}{2} \text{MAKNO}^2\right)^{(\gamma-1)/\gamma}$$

where WFC is the uncorrected fuel-flow rate, which is a function of EPR setting, altitude, and Mach number. It is obtained by evaluating a polynomial of EPRX, the coefficients of which depend on both altitude and Mach number.

If MFGR is set equal to 2, then the fuel-flow rate FDOT is obtained from the relations,

$$\text{FDOT} = \text{TSFC} * \text{THRST} * \sqrt{\theta}$$

$$\theta = T/T_o$$

where T_o is the temperature at sea level in degrees Kelvin. The thrust specific fuel consumption TSFC is a function of thrust (uncorrected for pressure ratio), Mach number, and altitude. It is obtained from tabulated data, stored in a three-dimensional array.

EPRIO

This subroutine reads in titles, one-dimensional array(s), and two-dimensional arrays, as specified by the values of KPRINT and LPRINT. It also prints out what was read-in if the value of IRPINT is other than zero. This is a satellite subroutine for CPMEPR. It is used for reading in and, if so desired, writing out the data set INP.ENGINE.

FUNCTION FBOUND (MACH)

This function evaluates the drag force D in pounds if IDRAG is set equal to 1, and the difference between maximum thrust (TMAX) and drag if IDRAG is equal to 2. If this difference is negative, that is, if drag exceeds thrust, then a value of 10^6 times the absolute value of this difference is returned to the calling program.

It first computes the lift coefficient CL by equating lift equal to weight, then determines the drag coefficient CD from known CL, and finally computes the drag force (see FCLIMB). To determine the maximum thrust, it calls the subroutine ENGEPR, setting EPR equal to its maximum value, EPRMAX.

Both D and TMAX are functions of Mach number and altitude H. The value of Mach number MACH and other quantities are passed through the argument of the function through the labeled common DRAGMN. Quantities passed through include the speed of sound A, and the pressure ratio RHO.

FCLIMB

This subroutine evaluates equation (11) for some given speed VTAS, energy E, and EPR setting EPRSET in normal operation or at idle-thrust setting if so specified. The denominator of equation (11) is the energy rate EDOT, obtained from evaluating equation (5).

The value of thrust in the EDOT equation and fuel-flow rate FF in the cost function are obtained by calling ENGEPR, or if idle thrust is specified, by extracting the appropriate value from the idle-thrust and idle-fuel arrays, FNIDL(.) and WFIDL(.), respectively. The thrust and fuel-flow rate are functions of altitude, Mach number, and, in the case of thrust, of EPR setting. Altitude and Mach number are computed from the relations,

$$H = E - \frac{(VTAS)^2}{2g} \qquad MACH = \frac{VTAS}{A}$$

where g is the acceleration of gravity and A is the speed of sound. The value of EPR setting is passed to this subroutine from the labeled common DRAGMN. The drag force D in the EDOT equation is obtained from the relations,

$$D = \frac{1}{2}\,(RHO)\,(VTAS)^2 C_D(MACH,CL) \qquad C_D = C_D(MACH,CL) \qquad C_L = \frac{W}{1/2\,(RHO)\,S\,(VTAS)^2}$$

where the drag coefficient C_D is obtained by calling the subroutine CDRAG. The values of altitude dependent quantities, A, RHO, and TEMPK are obtained by calling the subroutine AT62 with the known value of H as input. Atmospheric temperature TEMPK is used in both the ENGEPR subroutine and in the VALUE2 function.

Symbols

FNIDL idle thrust

RHO atmospheric pressure corrected for temperature

S wing reference area

WFIDL idle fuel-flow rate

FCLMB2

This subroutine minimizes the cost function (see eq. (11)) over speed or over both speed and thrust as specified.

First, the range of permissible speeds is determined. The lower and upper bounds of the speed range are derived from (1) thrust-drag consideration, (2) operational restraint, (3) structural limits, (4) ATC restriction, and (5) altitude ceiling, as discussed in the climb and descent optimization section.

Second, the permissible range of EPR settings for some fixed speed is also obtained. In the climb-profile optimization, it starts at the EPR value for which T = D, and ends at EPRMAX; in descent it ranges from 1.1 to EPR at T = D. Finally, the subroutine proceeds to determine the minimal value of the cost function. In the constrained-thrust case, it calls the function MINF to search for the minimum over the appropriate range with the value of EPR set to EPRMAX for climb and to idle EPR for descent. The cost function is evaluated by the subroutine FCLIMB.

In the free-thrust case, MINF is also used to determine the minimum of the cost function, searching first over the appropriate speed range, and then over the admissible range of EPR settings. The value of the free-thrust cost function is evaluated by the subroutine FTHRST. The value obtained from minimizing over speed is stored in COSTV; the value obtained from minimizing over thrust is stored in COSTPI. This process continues until the difference between the two is within EPSIL1, a small number. Since the minimum obtained becomes smaller in each successive search (as discussed in the section on climb/descent optimization), convergence is guaranteed. In the descent case, this cost is compared with that obtained from minimizing over the appropriate speed range at idle thrust. Then, the smaller of the two costs is selected as the minimum. The index IOPT is set equal to 1 for constrained thrust, to 2 for free thrust, and to 3 if both free thrust and a higher accuracy of speed and EPR setting are specified. In the third option, the speed accuracy is within 20/144 ft/sec and the EPR accuracy is within 0.02/144, where 20 is the speed range and 144 is the eleventh Fibonacci number.

FCOST

This subroutine evaluates equation (12) if ICOST is set equal to 1 and evaluates the fuel-flow rate FUEL if ICOST is set equal to 2, for some given Mach number MACH and altitude H. Cruise cost is expressed in pounds per nautical mile, and the fuel-flow rate is in pounds per hour. The fuel-flow rate is derived from keeping the aircraft in constant speed level flight at the given Mach number and altitude. It is computed in the subroutine TRIM1.

FDRAG

This subroutine computes the drag force D if IDRAG = 1 and the absolute value of T − D if IDRAG ≠ 1, for some given true airspeed VTAS and energy E. In the latter case, if T is less than D, then a value of 10^6 times the absolute value of T − D is returned.

The drag force is computed by equations discussed in the subroutine FCLIMB. The value of thrust T is computed in the subroutine ENGEPR, for known altitude, Mach number, and EPR setting.

The altitude and Mach number were computed from equations in FCLIMB, for known speed VTAS and energy E. The value of EPR setting is communicated to the subroutine from labeled common COST.

FOPT

This subroutine determines the minimum cruise cost and its corresponding cruise speed for some given altitude ALT. If ICOST = 1, equation (12) is used to evaluate the cost and if ICOST = 2, the fuel-flow rate FCOT is used.

In the first principal step, the given altitude is checked to ascertain whether it is below the operational ceiling, that is, if the thrust at maximum EPR setting is less than drag at the minimum drag speed. This is done in three substeps, the first of which is to compute the minimum drag speed by using FBOUND to evaluate the drag force D (i.e., with IDRAG set equal to 1) and by calling the subroutine MINF to search for the minimum value of drag over the Mach number range from 0 to 0.9. This value is stored in MINDRG and its corresponding speed in MACH. Next, compute TMAX, the value of thrust for the given altitude, known MACH and with EPR set equal to EPRMAX, by calling the subroutine ENGEPR. In the final substep, compare TMAX with MINDRG to ascertain if the former is greater than the latter. If so, continue; otherwise, exit the program.

After the altitude is checked, the lower and upper speed boundaries FA and FB are computed to define the valid speed range, where FA and FB are expressed as Mach numbers. The lower speed boundary is computed by determining the point between 0.1 Mach number and the minimum drag speed MINDRG at which TMAX equals D (see fig. 4(a)). Since TMAX and D are speed-dependent, they must be evaluated at different speeds. Specifically, FA is determined by calling the function FBOUND to evaluate the absolute value of TMAX - D (i.e., IDRAG = 2). The speed at which this quantity becomes zero is searched out by the subroutine MINF. The upper speed boundary, which lies between MINDRG and 0.9 Mach number, is computed in a similar fashion.

Finally, the subroutine FCOST is called to evaluate the cruise cost, and the minimum of all costs is searched out by the subroutine MINF over the valid speed range. The value of the appropriate minimum cruise cost is stored as the value of the function and the corresponding optimal cruise speed (for given altitude ALT) is stored in the array MACHOP(.). Specifically, the optimal cruise speed for minimizing equation (12) is stored in MACHOP(1), and for minimizing FDOT, in MACHOP(2).

FTHRST

This subroutine evaluates equation (11) for some given EPR setting EPR, Mach number MACH, true airspeed VTAS, altitude H, drag D, and weight W. The values of thrust T and fuel-flow rate FF are obtained by calling the subroutine ENGEPR.

It is a companion subroutine of FCLIMB. The value of EPR is the argument, and MACH, VTAS, H, D, and W are passed to the subroutine by the labeled commons CLIMB, ENERGY, DRAGMN, CLIMB, and COST, respectively.

FULEST

This subroutine estimates the initial cruise weight from known takeoff weight, initial energy, fuel cost, and time cost. The cruise weight is the difference between the takeoff weight and the climb fuel. The climb fuel is estimated from the following known quantities by an iterative process discussed in the section on weight estimation. It is estimated from the relation,

$$F_{up} = K_1(E_c - E_i)(1 + K_2'(T_c/F_c)(W_{to}/W_{ref})$$

$$= K_1(E_c - E_i)K_2(W_{to}/W_{ref})$$

where

E_c, E_i the cruise energy and initial energy

K_1 0.1079 for constrained thrust and 0.1130 for free thrust

K_2 4.6849E-6 for constrained thrust and 6.1463E-6 for free thrust

T_c, F_c time cost and fuel cost

W_{to}, W_{ref} takeoff weight and some reference takeoff weight from which the value of the constants K_1, K_2 was derived

Steps for determining the initial cruise weight are (1) set W_c, the cruise weight, equal to takeoff weight; (2) compute the cruise energy ECRUZ corresponding to the cruise weight W_c by calling the subroutine WLEFHV with the control index I set equal to 1; (3) use the given relation to compute the climb fuel F_{up}; (4) subtract F_{up} from the takeoff weight to arrive at the cruise weight W_c; and (5) test this weight against the one obtained in the previous iteration. If they are within 100 lb of each other, the estimation is complete; if not, return to step (2).

ICLOCK

This subroutine resolves TIME in seconds into hours IHR, minutes IMIN, and seconds ISEC. For example, 4521 sec is resolved into 1 hr 15 min, and 21 sec.

JTRUNC

This subroutine finds the last element of a monotonically decreasing series within an unimodal series.

LSQPOL

This subroutine makes a least-squares polynomial fit for y as a function of x. The arrays x (·), y (·) contain points of the independent variable, and the corresponding value of the function. The total number of points in the set is specified by N, the degree of the polynomial to be fitted is specified by M − 1 (e.g., M = 3 for quadratic fit), and the coefficients of the polynomial are stored in the array B (·) in the following order,

$$y = B(3)x^2 + B(2)x + B(3)$$

MATINV

This subroutine takes the inverse of a matrix A and stores the result in A. The dimension of the matrix is specified by M.

MINF

This subroutine minimizes the unimodal funtion F(X) so that x is within 1/144 of the given interval (BX - AX) by a Fibonacci search, where 144 is the 11th Fibonacci number.

Symbols

AX, BX — lower and upper limits of X

F( ) — the unimodal function to be minimized; this function is specified in the calling program

FIBONO(.) — ratio of Fibonacci numbers f_i/f_{11}, i = 1,9

IPRINT — if = 0, no print; ≠ 0, print x and F(x) in each step of the search

X — the value of x at which the unimodal function F(x) is at its minimum

MINF2

This subroutine minimizes the unimodal function F(x) so that x is within 1/55 of the given interval (BX - AX), using a Fibonacci search, where 55 is the ninth Fibonacci number.

Symbols

FIBON(.) — ratio of Fibonacci numbers f_i/f_g, i = 1, 7 (see MINF for explanation of other symbols)

PCCOMP

This subroutine contains the logic for (1) exiting the program when the desired range R_d is less than R_{min} or when it is within a small distance of R, the ground-track distance subtended by the synthesized trajectory; (2) setting up a flag for splicing in a cruise segment subtending a cruise distance $d_c = R_d - (d_{up} + d_{dn})$ if R_d is greater than R^*, and (3) computing the value of $\%_\lambda$ for synthesizing the desired trajectory if R_d is between R_{min} and R^*. The symbols d_{up} and d_{dn} stand for climb distance and descent distance, respectively.

In the first case, either the desired range cannot be synthesized or the desired trajectory has been synthesized. In the second case, the index ISPLIZ is set equal to zero. This value is communicated to the subroutine VOPTRJ through the arguments of the two subroutines; the next execution step is transferred to VOPTRJ. The subroutine VOPTRJ contains the required logic to complete the synthesis. In the third case, the value of $\%_\lambda$ is computed to start the iteration until the desired range trajectory is synthesized.

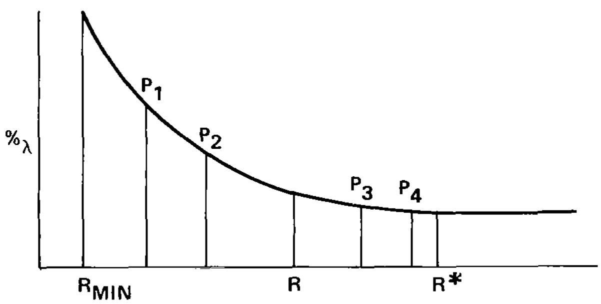


Figure 17.- Range as a function of $\%_\lambda$.

Figure 17 shows the relation between $\%_\lambda$ and range. Consider the region between R_{min} and R^*. In the short-range region, midrange region, and long-range region, $\%_\lambda$ decreases as the reciprocal of R^2, as the reciprocal of R, and linearly with R, respectively. These relationships are represented, respectively by the following equations,

$$\%_\lambda = (R - R_1)(\%_\lambda 2 - \%_\lambda 1)/(R2 - R1) \tag{A1}$$

$$\%_\lambda = A/R + B \ , \quad \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} 1/R1 & 1 \\ 1/R2 & 1 \end{pmatrix}^{-1} \begin{pmatrix} \%_\lambda 1 \\ \%_\lambda 2 \end{pmatrix} \tag{A2}$$

$$\%_\lambda = A/R^2 + B \ , \quad \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} 1/R_1{}^2 & 1 \\ 1/R_2{}^2 & 1 \end{pmatrix}^{-1} \begin{pmatrix} \%_\lambda 1 \\ \%_\lambda 2 \end{pmatrix} \tag{A3}$$

where $(R_1, \%_\lambda 1)$ and $(R_2, \%_\lambda 2)$ are range and percent points obtained in the previous iterations. To minimize the number of iteration steps, it is important to take advantage of the curve to determine the required value of $\%_\lambda$ for synthesizing the desired trajectory. This is done by selecting two points closest to R_d and by implementing the logic to select the appropriate shape of the curve in regions under consideration.

In the curve of figure 17, P_1, P_2, P_3, and P_4 are points, if available, obtained from previous iterations. The first two are for distances, subtended by trajectories, that are shorter than the desired range; the remaining two are for distances that are longer than R_d. Only two of these points, which can best approximate the local fit of the curve, will be used. They will be denoted by P_a and P_b. The criteria for their selection are (1) use P_2 and P_3 in the first iteration; (2) if $\delta_3 \le \delta_1 + \Delta R$, use P_3; otherwise use P_1 where $\delta_i = |R_d - R_i|$, i = 1, 4, and ΔR is some specified distance (ΔR is set equal to 10 nm if $\%P_2$ is less than 2, and to 20 nm otherwise); (3) if $\delta_4 \le \delta_2 + \Delta R$, use P_2; otherwise use P_4.

The criteria for selecting the proper local shape of the curve are

1. Use linear interpolation in the first iteration if both δ_2 and δ_3 are greater than 10 n. mi.

2. Use linear interpolation-extrapolation if $|\%_{P_a} - \%_{P_b}| \le 2$ (see eq. (A1)).

3. Use reciprocal fit if $\%_{P_2} < 2$ (see eq. (A2)).

4. Use reciprocal of R^2 fit if $\%_{P_2} \ge 2$ (see eq. (A3)).

5. If in successive iteration steps, the tests pass either criterion (2) or criterion (3) repeatedly, then alternate between these two criteria.

PILIMT

This subroutine evaluates the absolute value of T - D for some given EPR setting EPR, altitude H, Mach number MACH, and drag D. The value of thrust T is computed by calling the subroutine ENGEPR with known H, MACH, and EPR as inputs. The value of D is passed to this subroutine by the labeled common CLIMB.

This subroutine is used in conjunction with MINF to define the thrust boundary for climb and descent optimization. The value of $|T-D|$ is multiplied by 10^6 if it is outside the boundary (see fig. 4). That is, a very heavy penalty is imposed on the function if $T < D$ for climb and $T > D$ for descent.

POLYE1

This function evaluates the polynomial

$$Y = b(1) + b(2)X + b(3)X^2 + \ldots b(M)X^{m-1}$$

SERCHD

This subroutine searches the index ℓ such that $TX(\ell) \geq X \geq TX(\ell + 1)$ and returns both ℓ and pf where

$$pf = \frac{TX(\ell) - X}{TX(\ell) - TX(\ell + 1)}$$

and TX(.) is a monotonically decreasing array.

SERCH1

This subroutine searches the index ℓ such that

$$TX(\ell) \leq X \leq TX(\ell + 1)$$

and returns both ℓ and pf where

$$pf = \frac{X - TX(\ell)}{TX(\ell + 1) - TX(\ell)}$$

and TX(.) is a monotonically increasing array.

TRIM1

This is a general subroutine for keeping the aircraft in a certain flight condition, depending on the value of MODE and on the value of ICNTRL if MODE = 1. MODE = 1 means that the EPR setting EPR is specified; MODE = 2 that the flightpath angle γ is specified; and MODE = 3 that both EPR and flightpath angle are specified. There are three submodes for the first option. They are specified by the value of ICNTRL. ICNTRL = 1 specifies that the aircraft be flown at constant true airspeed, ICNTRL = 2, at constant MACH number, and ICNTRL = 3, at constant indicated airspeed. In all modes, the flightpath angle is constant. It is determined from solving the $\dot{\gamma}$ and the V equations simultaneously in mode 1 and is specified in both modes 2 and 3.

The value of the mode to be used for synthesizing the optimal trajectory is 2, and the description for the remainder of this subroutine will be confined to this mode only.

If MODE = 2, this subroutine computes the angle of attack ALPHA (α), and the EPR setting EPR to keep the aircraft in trim, that is, constant-speed level flight, for some given altitude h and true airspeed VTAS.

First, α is determined by combining the following relations,

$$(T \sin \alpha + L)\cos \phi - W = 0 \tag{A4}$$

$$L = \frac{1}{2} \rho S V^2 C_L \tag{A5}$$

$$C_L = CL(basic) + CL(\alpha)\alpha + CLO + CL(gear) \tag{A6}$$

$$CL(basic) = b(1) + b(2)\alpha + b(3)\alpha^2 \tag{A7}$$

to arrive at the quadratic

$$A\alpha^2 + B\alpha + C = 0 \tag{A8a}$$

where

$$A = \frac{1}{2} \rho S V^2 \cos \phi \ b(3) \tag{A8b}$$

$$B = \left\{\frac{1}{2} \rho S V^2 [b(2) + CL(\alpha)] + T\right\} \cos \phi \tag{A8c}$$

$$C = \frac{1}{2} \rho S V^2 \cos \phi [b(1) + CLO, + CL(gear)] - W \tag{A8d}$$

Equation (A4) is obtained by setting γ and $\dot{\gamma} = 0$ in the γ equation. Equation (A5) is the relation between the lift force and the lift coefficient. Equation (A6) is the same lift coefficient equation as that stated in CLIFT. And equation (A7) defines CL(basic) as a polynomial in α. The coefficients of the polynomial are obtained by fitting the curves in pages 2.1-2.6 of reference 7.

The value of thrust T in equation (A8a) is obtained by assuming some value of EPR setting and then calling the subroutine ENGEPR to generate the appropriate value of thrust T. The values of CLO, CL, and CL(gear) are constants for flaps up.

Next, the exact value of EPR to keep the aircraft in trim, for the α obtained in equation (A8a) is determined by setting $\dot{V} = 0$ and $\gamma = 0$ in the $\dot{V}$ equation, that is,

$$T \cos \alpha - D = 0 \tag{A9}$$

where D, the drag force in pounds, is computed by calling the subroutine CDRAG.

This subroutine iterates between equations (A8a) and (A9) until a pair of (α, T) is obtained that are consistent with both equations.

Symbols

γ flightpath angle, deg

$\dot{\gamma}$ rate of change of flightpath angle

T thrust, lb

α angle of attack, deg

ϕ bank angle, deg

W weight of aircraft, lb

ρ air density, lb sec^2/ft^4

S wing reference area, ft^2

V true airspeed, ft/sec

L lift force, lb

CLO, CL(gear), CL216(.) see CLIFT

INCRUZ see ENGEPR

UPDOWN

This subroutine generates the climb profile if ICLIMB = 1 and the descent profile is ICLIMB = 2.

The climb profile is generated starting with initial energy E_o, incrementing with ΔE, and finishing at E_c. The descent profile is generated in the same manner, using final-energy E_f as the starting point. The initial energy is specified by the initial speed V_o and altitude h_o and, similarly, the final energy by the final speed V_f and final altitude h_f. The incremental energy is 500 energy feet except when the energy level is within 3,000 energy feet of the cruise energy E_c, at which point it is reduced to 250 ft. The cruise energy is computed in the subroutine FULEST (see MAINPROG, step (6)) for climb, and in the VOPTRJ subroutine in descent. It is transferred to this subroutine through the argument.

At each energy level, the subroutine FCLMB2 is called to minimize the cost function (eq. (12)). Equations (16) through (19) are used to compute the time duration, distance traversed, flightpath angle, and fuel used for the profile, starting from the initial/final energy to the energy level under consideration. These and other quantities are printed out in table 7 for the climb profile and in table 8 for the descent profile.

The speed available from these profiles is either in true airspeed or Mach number. Another frequently used quantity, the indicated airspeed, is converted from the Mach number by the relation,

$$\mathrm{VIAS} = \sqrt{\frac{2}{\rho_{s\ell}}}\sqrt{P\left(1 + \frac{\gamma - 1}{2} m^2\right)^{\gamma-1/\gamma} - 1} \qquad \text{(A10)}$$

It is derived from combining the relations,

$$\text{VIAS} = \sqrt{2(P_o - P)/\rho_{s\ell}} \tag{A11}$$

$$P_o = P\left(1 + \frac{\gamma - 1}{2} m^2\right)^{\gamma-1/\gamma} \tag{A12}$$

Since $\rho_{s\ell} = 0.002378$ slug/ft^3 (lb sec^2/ft^4) and $\gamma = 1.4$, equation (A10) can be simplified to

$$\text{VIAS} = 29\sqrt{P(1 - 0.2\ m^2)^{3.5} - 1} \tag{A13}$$

Symbols

m Mach number

P_o stagnation pressure

P static pressure which is a function of altitude

γ ratio of specific heats (1.4)

$\rho_{s\ell}$ air density at sea level and standard atmospheric pressure (29.92 in. of Hg) and temperature (15° C)

VALUE2

This subroutine interpolates (or under certain conditions, extrapolates) the value of a function of two variables stored in the two-dimensional array EPRMAX(.). This array may be viewed as a table with rows and columns. The value of the function depends on the two variables TEMPA and H. The incremental values of TEMPA, stored in the array TEMP(.), are used for indexing the rows, and the incremental values of H, stored in the array ALT24(.), are used for indexing the columns.

The subroutine will interpolate a point if it is within the table. If it is outside the table and I = 1, it will truncate the value to the nearest point in the table. That is, if TEMPA is less than TEMP(1), then the subroutine assumes the point to be in the first row. And if TEMPA is greater than the last element of TEMP(.), it assumes the point to be in the last row. Similarly, if H is less than ALT24(1) or greater than the last element of ALT24(.), it will assume the point to be in the first column and the last column, respectively.

If the point is outside the table and I ≠ 1, the subroutine will truncate the point to the nearest column. If in addition, TEMPA is less than TEMP(1), it will call XTRPL1 to extrapolate a number that is beyond the first row. Likewise, it will call XTRPL2 to extrapolate a number that is beyond the last row, if TEMPA is greater than the last element of TEMP(.).

VOPTRJ

This subroutine generates the quantities shown in table 9. They include the fuel used, distance traversed, time duration, cost in dollars, and in dollars per nautical mile for the climb profile, the descent profile, the cruise segment, and the overall trajectory. Other cruise quantities are the weight, cost, energy, altitude, and speed at the beginning and end of cruise.

The fuel used, distance traversed, time duration, etc., were evaluated at the cruise energy ECRUZ.

In the free-thrust case, the cruise energy is obtained by evaluating the energy level at which the sum of the Hamiltonian for climb and that for descent goes to zero. The Hamiltonian for climb and for descent for each energy level was obtained from minimization. They appear in column (12) in tables 7 and 8 and their sum in column (13) in table 8. If column (13) does not contain this energy level, i.e., the elements of this column are all positive, then the last energy level shown in the table, or the energy level corresponding to the smallest value of the sum of the two Hamiltonians, will be used.

In the constrained-thrust case, the cruise energy is evaluated in step (6) in MAINPROG.

In the free-thrust (short and intermediate ranges) case, there is no cruise segment, and the cruise quantities are the same for both the beginning and the end of cruise. In all other cases, they differ because of fuel burned during cruise. The amount of fuel burned is computed by the use of equation (28). The cruise fuel is expressed as the product of fuel-flow rate and cruise distance divided by groundspeed. Both the fuel-flow rate and the groundspeed were evaluated at ECRUZ. The cruise distance is computed by the appropriate equation, depending on whether the aircraft is cruising below or at optimal cruise altitude.

If the aircraft cruises below optimal cruise altitude (i.e., constrained-thrust, short- and intermediate-range trajectories), the cruise distance is computed by the use of equation (22) evaluated at ECRUZ. The denominator of equation (22) is $d\lambda/dE$; it is expressed as a polynomial in E. The coefficient of the polynomial is obtained by curve fitting λ vs E and then taking the derivative. There are a number of $d\lambda/dE$ curves, each corresponding to a specific cruise weight. The data for each λ vs E curve were stored in the array LLCRUZ(., IW) and EECRUZ(., IW), respectively. The integer IW is for indexing the cruise weight. These arrays are parts of the cruise table, with each value of IW referring to a specific page in the table. For example, IW = 1 refers to a cruise weight of 150,000 lb.

The cruise cost λ, is fitted as a quadratic in E.

$$\lambda = c_1 E^2 + c_2' E + c_3 \tag{A14}$$

Taking the derivative with respect to E:

$$d\lambda/dE = 2c_1E + c_2' \tag{A15}$$

This derivative is used to evaluate the length of the cruise segment d_c. Figure 18 shows a typical λ vs E and $d\lambda/dE$ vs E curve. At the optimal energy E^*, $d\lambda/dE = 0$. Due to inaccuracy in curve fitting, it is possible for the $d\lambda/dE$ curve to be displaced, as shown by the dashed line. In this case, the cruise distance can be negative. For example, if d_c is evaluated at E_c as shown, then $d\lambda/dE$ is positive. Since $I_{up} + I_{dn}$ is also positive for E_c less than E^*, the value d_c is negative. To prevent this from happening, the constant c_2' is adjusted so that $d\lambda/dE$ is zero at E^* by the relation,

$$c_2 = -2c_1E^* \tag{A16}$$

This is equivalent to refining the accuracy of the E-intercept of the $d\lambda/dE$ curve.

In order that the units be consistent in equation (A15), this derivative must be expressed in dollars per nautical mile squared. Since the cruise energy E is expressed in feet, the denominator of the derivative must be converted into nautical miles. Therefore, equation (A10) can be written as

$$\frac{d\lambda}{dE} = AE + B \tag{A17}$$

where $A = 2(6080)c_1$, $B = 6080c_2$, and A and B are coefficients for each cruise weight. They are stored in the arrays DLLDEE(1,IW) and DLLDEE(2,IW), respectively. Some typical values of A and B are shown in table 6, columns (2) and (3). Each set corresponds to the cruise weight shown in column (1). Since no aircraft cruises below 20,000 ft altitude, only points between 20,000 ft altitude and the optimal cruise altitude were used for the curve fitting.

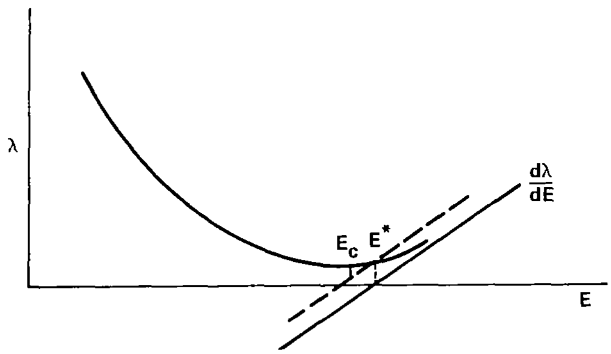


Figure 18.- Cruise cost, and $d\lambda/dE$ vs energy.

If the aircraft cruises at optimal cruise altitude (i.e., long-range flight for both the free- and constrained-thrust cases), then the distance is evaluated by the relation

$$d_c = R - d_{up} + d_{dn}$$

Symbols

DCLMB, DDOWN, DCRUZ, TDIST	distance covered by the climb profile, descent profile, cruise segment, and the overall trajectory
EFCRUZ, EFFCNZ	cruise efficiency and overall efficiency (for the whole trajectory)
ECRUZ, ECRUZI, ECRUZF	cruise energy, energy at beginning and end of cruise
FCLIMB, FDOWN, FCRULB, TFUEL	climb fuel, descent fuel, cruise fuel, total fuel
HCRUZ, HCRUZI, HCRUZF	cruise altitude, altitude at beginning and end of cruise
IOPARM, ISPLIZ	see MAINPROG
LAMBDA, LAMBDI, LAMBDF	cruise cost, and cost at beginning and end of cruise
MACH, MACHI, MACHF	Mach number at cruise, Mach number at beginning and end of cruise
SSCOST(.)	sum of I_{up} and I_{dn}
TCLMB, TDOWN, TCRUZ, TTIME	time duration for climb, descent, cruise, and for the overall trajectory
VGKNT, VGKNTI, VGKNTF	groundspeed at cruise, and groundspeed at beginning and end of cruise
WCRUZ, WCRUZI, WCRUZF	cruise weight, and weight at beginning and end of cruise
WLNDG	landing weight

WATEST

This subroutine is used to estimate the landing weight upon the completion of synthesizing the climb profile.

In the free-thrust case, there is no cruise segment and the landing weight is obtained by subtracting the climb fuel FCLIMB and the descent fuel FDOWN from the takeoff weight WTO. The climb fuel is interpolated from the array FFCLMB(.), which stores the fuel burned for different energy heights in the climb profile. The descent fuel is estimated from a polynomial in $\%_{\lambda}$. Its coefficients were obtained by fitting the appropriate curve derived from previously synthesized trajectories. They are stored in the array DNFUEL(.).

In the constrained-thrust case, there is a cruise segment; therefore, the fuel burned during cruise must also be subtracted from the takeoff weight to arrive at the landing weight. Equation (29) is used for computing the cruise fuel, which is the product of fuel-flow rate and cruise distance divided by groundspeed. The fuel-flow rate and the groundspeed were computed by calling the subroutine WLEFHV with index I set equal to 2. The cruise distance is estimated by a polynomial in $\%_\lambda$. The polynomial coefficients were also obtained by fitting the appropriate curve derived from previously synthesized trajectories. They are stored in the array DCOEF(.).

WIND

This subroutine computes the component of wind velocity projected along the principal axis of the aircraft. The wind profile itself is a function of altitude. Inputs to this program are altitude in feet H, the aircraft heading in degrees PSIA, and the wind profile stored in the arrays HWIND(.), VW(.), and PSIW(.). Output from this program is VWA, the required wind component.

WINDIN

This subroutine reads in the wind profile, the magnitude and direction of wind, and also computes the wind shear for every 2,000 ft of altitude. The wind magnitude from input is in knots; the program converts it to feet per second and stores it in the VW(.) array. The wind direction is in 10° increments. The program converts it into degrees and stores it in PSIW. The altitude corresponding to the wind magnitude and direction is generated internally and stored in the HWIND(.) array.

The x, y components of wind are computed by the relations,

$$\partial W_x/\partial h = \Delta W_x/\Delta h \tag{A18}$$

$$\partial W_y/\partial h = \Delta W_y/\Delta h \tag{A19}$$

$$W_x = VW(.)\cos(PSIW(.)) \tag{A20}$$

$$W_y = VW(.)\sin(PSIW(.)) \tag{A21}$$

The x, y components of the wind and also the x, y components of wind shear, all of which are altitude dependent, are stored in the arrays WX(.), WY(.), WINDE(., 2), WINDE(., 3), respectively. These arrays are indexed to altitude.

WINMOD

This subroutine computes the component of wind shear along the principal axis of the aircraft for some given altitude ALT (in feet) by the relation

$$\partial W_a/\partial h = (\partial W_x/\partial h)\cos\psi + (\partial W_y/\partial h)\sin\psi \qquad \text{(A22)}$$

where ψ is the aircraft heading, $\partial W_x/\partial h$, $\partial W_y/\partial h$ are x, y components of wind shear stored in the arrays WINDE(., 2), WINDE(., 3), respectively. The resulting component of wind shear is stored in AKW.

WLEFHV

This subroutine computes various cruise quantities for some given cruise weight WCRUZ and percentage over the optimal cruise cost $\%_\lambda$, depending on the value of the index I. These quantities include the cruise cost LAMBDA, cruise energy ECRUZ, cruise altitude HCRUZ, fuel-flow rate FCRUZ, and groundspeed (in knots) VGKNT. For I = 1, LAMBDA and ECRUZ are computed; for I = 2, FCRUZ, HCRUZ, VCKNT are computed; for I = 3, LAMBDA, ECRUZ, HCRUZ, FCRUZ, and VGKNT are computed; for I = 4, HCRUZ is computed.

Symbols

IW, pfw (see explanation for 1 and pf in SERCH1)

WRITE1

This subroutine computes the cost and unit cost for flying a specified segment of the trajectory. The segment can be climb, descent, cruise, or the overall trajectory, as specified in the calling program. Also, it resolves the time given in seconds into hours, minutes, and seconds. Finally, it prints out the description of the segment (i.e., climb, cruise, descent, total), the fuel used, the distance traversed, the time duration, in hr:min:sec, the cost, and the unit cost. It is a satellite subroutine for VOPTRJ.

XTRPL1

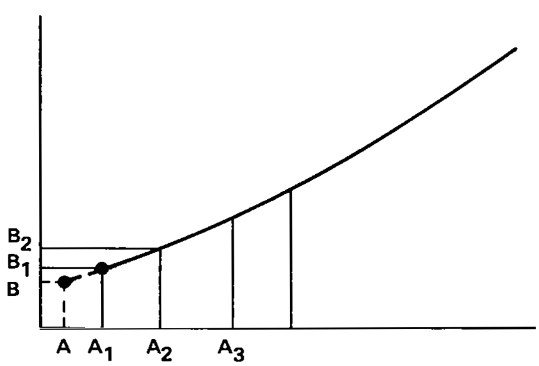
This subroutine extrapolates a point below the lowest point in some given array. Figure 19(a) shows the lowest point as (A1, B1) where the value of B, corresponding to A is desired. It is obtained by the rela relation

$$\frac{B2 - B}{A2 - A} = \frac{B2 - B1}{A2 - A1} \qquad \text{(A23a)}$$

or

$$B = B_2 - \frac{A2 - A}{A2 - A1}(B2 - B1) \qquad \text{(A23b)}$$

This is a satellite subroutine for VALUE2.



(a) Lower end.

Figure 19.- Extrapolation.

XTRPL2

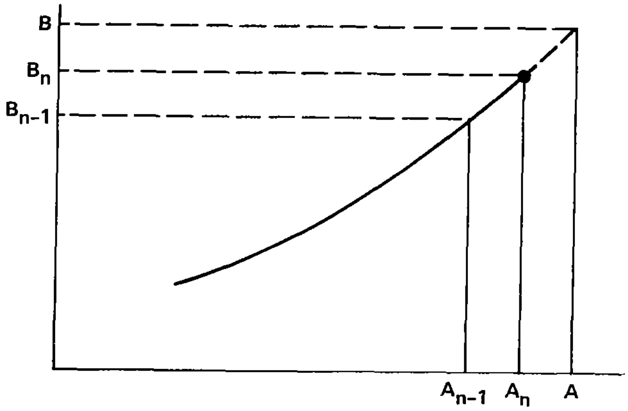
This subroutine extrapolates a point beyond the highest point in some given array. Figure 19(b) shows the highest point as (An, Bn) where the value of B, corresponding to A is desired. It is obtained by the relation

$$\frac{B - B_{n-1}}{A - A_{n-1}} = \frac{Bn - B_{n-1}}{An - A_{n-1}} \tag{A24a}$$

or

$$B = B_{n-1} + \frac{A - A_{n-1}}{An - A_{n-1}} (Bn - B_{n-1}) \tag{A24b}$$

This is a satellite subroutine for VALUE2.



(b) Upper end.

Figure 19.- Concluded.

REFERENCES

1. Erzberger, Heinz; and Lee, Homer Q.: Characteristics of Constrained Optimum Trajectories with Specified Range. NASA TM-78519, 1978.

2. Erzberger, Heinz; McLean, John D.; and Barman, John F.: Fixed-Range Optimum Trajectories for Short-Haul Aircraft. NASA TN D-8115, 1975.

3. Bochem, J. H.; and Mossman, D. C.: Simulator Evaluation of Optimal Thrust Management/Fuel Conservation Strategies for Airbus Aircraft and Short-Haul Routes. Final report, May 1977-April 1978, June 1978.

4. Cooper L.; and Steinberg, D.: Introduction to Methods of Optimization. W. B. Saunders Co., Philadelphia, PA, 1970, pp. 147-151.

5. Daniels, Glenn E.; and Smith, Orvel E.: Scalar and Component Wind Correlations between Altitude Levels for Cape Kennedy, Florida, and Santa Monica, California. NASA TN D-3815, 1968.

6. U.S. Standard Atmosphere Supplements, 1962. Prepared under the sponsorship of Environmental Science Services Administration, NASA, and U.S. Air Force.

7. Aerodynamic Data for the 727 Flight Simulator. Document No. D6-8095, Boeing Company, Seattle, Washington.

8. Specific Operating Instructions, JT8D-7A. Pratt and Whitney, A Division of United Technology, July 15, 1975.